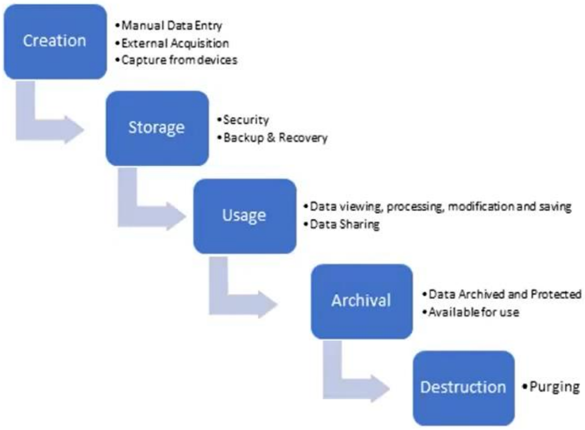
Life Cycle of Data

- The data lifecycle represents all of the stages of data throughout its life from its creation for a study to its distribution and destruction.



- **Data Collection/Creation**: In this phase, data comes into an organization, usually through data entry, acquisition from an external source or signal reception, such as transmitted sensor data.
- **Data Storage**: Once data has been created within the organization, it needs to be stored and protected, with the appropriate level of security applied. A robust backup and recovery process should also be implemented to ensure retention of data during the lifecycle.

- **Data Usage:** During the usage phase of the data lifecycle, data is used to support activities in the organization. Data can be viewed, processed, modified and saved. Data may also be made available to share with others outside the organization.
- **Data Archival:** Data Archival is the process of removing data from active production environment and keeping copy of data so that it can be used again in an active production environment, if needed.

- **Data Destruction**: Data destruction or purging is the removal of every copy of a data item from an organization. It is typically done from an archive storage location. If we want to save all data forever, it's not feasible. Storage cost and compliance issues create pressure to destroy data no longer need.

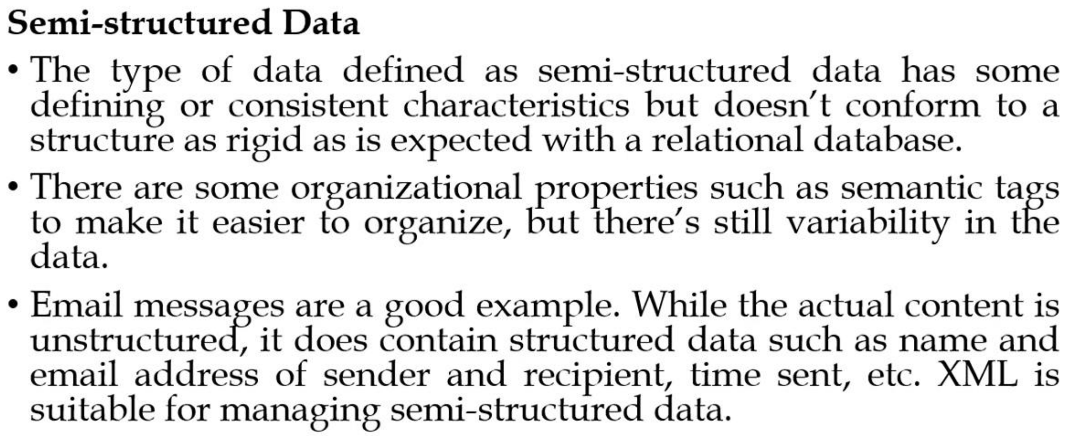
Types of Data

Unstructured Data

- Data that cannot be contained in a row-column database unstructured data and doesn't have an associated data model. The lack of structure made unstructured data more difficult to search, manage and analyze, which is why companies have widely discarded unstructured data, until the recent proliferation of AI and machine learning algorithms made it easier to process.
- Examples of unstructured data include photos, video and audio files, text files, social media content, open-ended survey responses etc. Instead of relational databases, unstructured data is usually stored NoSQL databases and data warehouses.

Structured Data

- Data that is the easiest to search and organize, because it is usually contained in rows and columns and its elements can be mapped into fixed pre-defined fields, is known as structured data. Structured data follows a relational data model.
- Relation Databases and SQL is suitable for managing structured data.



Semi-structured Data

- The type of data defined as semi-structured data has some defining or consistent characteristics but doesn't conform to a structure as rigid as is expected with a relational database.
- There are some organizational properties such as semantic tags to make it easier to organize, but there's still variability in the data.
- Email messages are a good example. While the actual content is unstructured, it does contain structured data such as name and email address of sender and recipient, time sent, etc. XML is suitable for managing semi-structured data.

Data warehouse and data warehousing

A **data warehouse** is designed to run query and analysis on historical **data** derived from transactional sources for business intelligence and **data** mining purposes. **Data warehousing** is used to provide greater insight into the performance of a company by comparing **data** consolidated from multiple heterogeneous sources.

A Data Warehousing (DW) is process for collecting and managing data from varied sources to provide meaningful business insights. A Data warehouse is typically used to connect and analyze business data from heterogeneous sources. The data warehouse is the core of the BI system which is built for data analysis and reporting.

Types of Data Warehouse

Three main types of Data Warehouses (DWH) are:

1. Enterprise Data Warehouse (EDW):

Enterprise Data Warehouse (EDW) is a centralized warehouse. It provides decision support service across the enterprise. It offers a unified approach for organizing and representing data. It also provide the ability to classify data according to the subject and give access according to those divisions.

2. Operational Data Store:

Operational Data Store, which is also called ODS, are nothing but data store required when neither Data warehouse nor OLTP systems support organizations reporting needs. In ODS, Data warehouse is refreshed in real time. Hence, it is widely preferred for routine activities like storing records of the Employees.

3. Data Mart:

A data mart is a subset of the data warehouse. It specially designed for a particular line of business, such as sales, finance, sales or finance. In an independent data mart, data can collect directly from sources.

Difference between Operational Database and Data Warehouse

Operational Database	Data Warehouse
Operational systems are designed to support high-volume transaction processing.	Data warehousing systems are typically designed to support high-volume analytical processing (i.e., OLAP).
Operational systems are usually concerned with current data.	Data warehousing systems are usually concerned with historical data.
Data within operational systems are mainly updated regularly according to need.	Non-volatile, new data may be added regularly. Once Added rarely changed.
It is designed for real-time business dealing and processes.	It is designed for analysis of business measures by subject area, categories, and attributes.
It is optimized for a simple set of transactions, generally adding or retrieving a single row at a time per table.	It is optimized for extent loads and high, complex, unpredictable queries that access many rows per table.
It is optimized for validation of incoming information during transactions, uses validation data tables.	Loaded with consistent, valid information, requires no real-time validation.
It supports thousands of concurrent clients.	It supports a few concurrent clients relative to OLTP.

Operational systems are widely process-oriented.	Data warehousing systems are widely subject-oriented
Operational systems are usually optimized to perform fast inserts and updates of associatively small volumes of data.	Data warehousing systems are usually optimized to perform fast retrievals of relatively high volumes of data.
Data In	Data Out
Less Number of data accessed.	Large Number of data accessed.
Relational databases are created for on-line transactional Processing (OLTP)	Data Warehouse designed for on-line Analytical Processing (OLAP)

A multidimensional data model,

Multidimensional data model stores data in the form of data cube. Mostly, data warehousing supports two or three-dimensional cubes.

When data is grouped or combined in multidimensional matrices called Data Cubes. The data cube method has a few alternative names or a few variants, such as "Multidimensional databases," "materialized views," and "OLAP (On-Line Analytical Processing)."

A data cube allows data to be viewed in multiple dimensions. A dimension are entities with respect to which an organization wants to keep records. For example, in store sales record, dimensions allow the store to keep track of things like monthly sales of items and the branches and locations.

A multidimensional database helps to provide data-related answers to complex business queries quickly and accurately.

Data warehouses and Online Analytical Processing (OLAP) tools are based on a multidimensional data model. OLAP in data warehousing enables users to view data from different angles and dimensions.

Schemas for Multidimensional Data Model are: -

Star Schema

Snowflakes Schemes

Fact Constellations Schema

Difference between OLTP and OLAP

OLAP (Online analytical processing)	OLTP (Online transaction processing)
Consists of historical data from various Databases.	Consists only operational current data.
It is subject oriented. Used for Data Mining, Analytics, Decision making,etc.	It is application oriented. Used for business tasks.
The data is used in planning, problem solving and decision making.	The data is used to perform day to day fundamental operations.
It reveals a snapshot of present business tasks.	It provides a multi-dimensional view of different business tasks.
Large amount of data is stored typically in TB, PB	The size of the data is relatively small as the historical data is archived. For ex MB, GB
Relatively slow as the amount of data involved is large. Queries may take hours.	Very Fast as the queries operate on 5% of the data.
It only need backup from time to time as compared to OLTP.	Backup and recovery process is maintained religiously
This data is generally managed by CEO, MD, GM.	This data is managed by clerks, managers.
Only read and rarely write operation.	Both read and write operations.

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 −

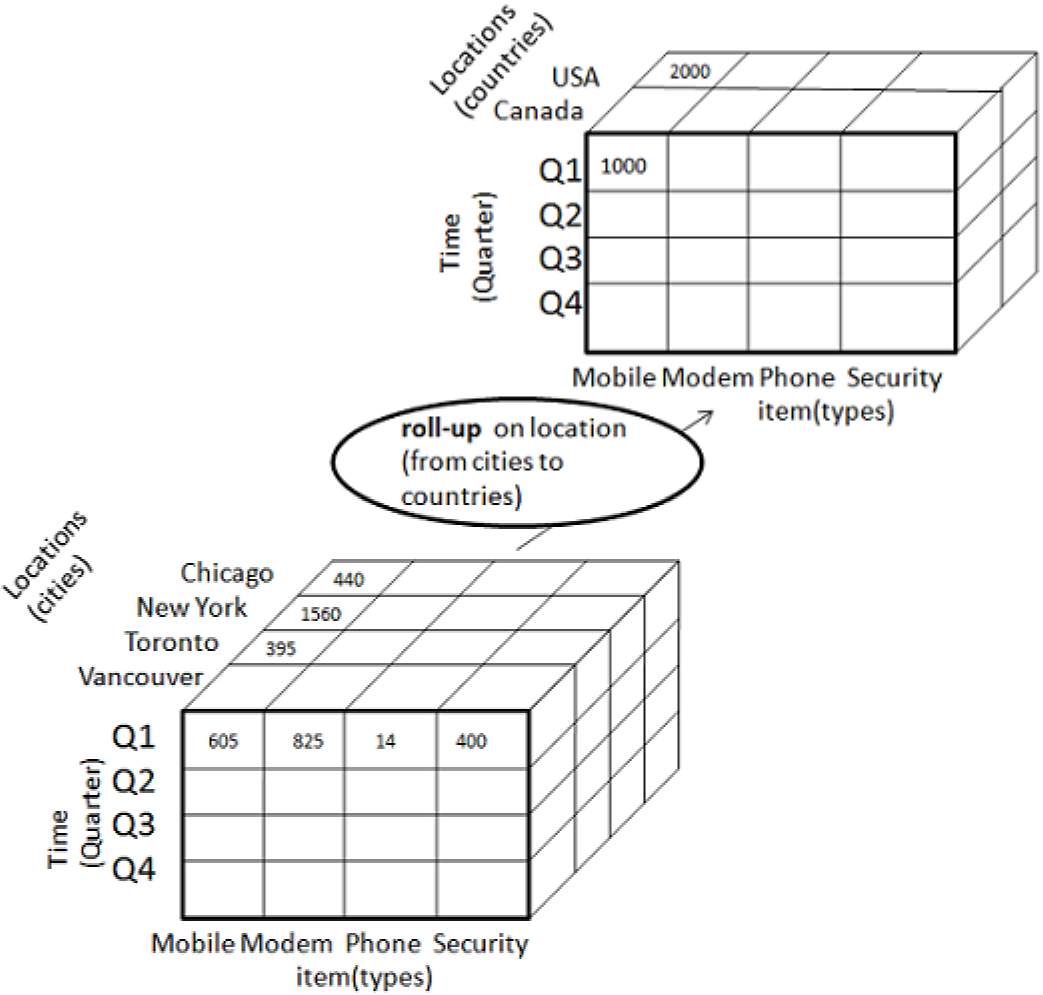
- Roll-up
- Drill-down
- Slice and dice
- Pivot (rotate)

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.



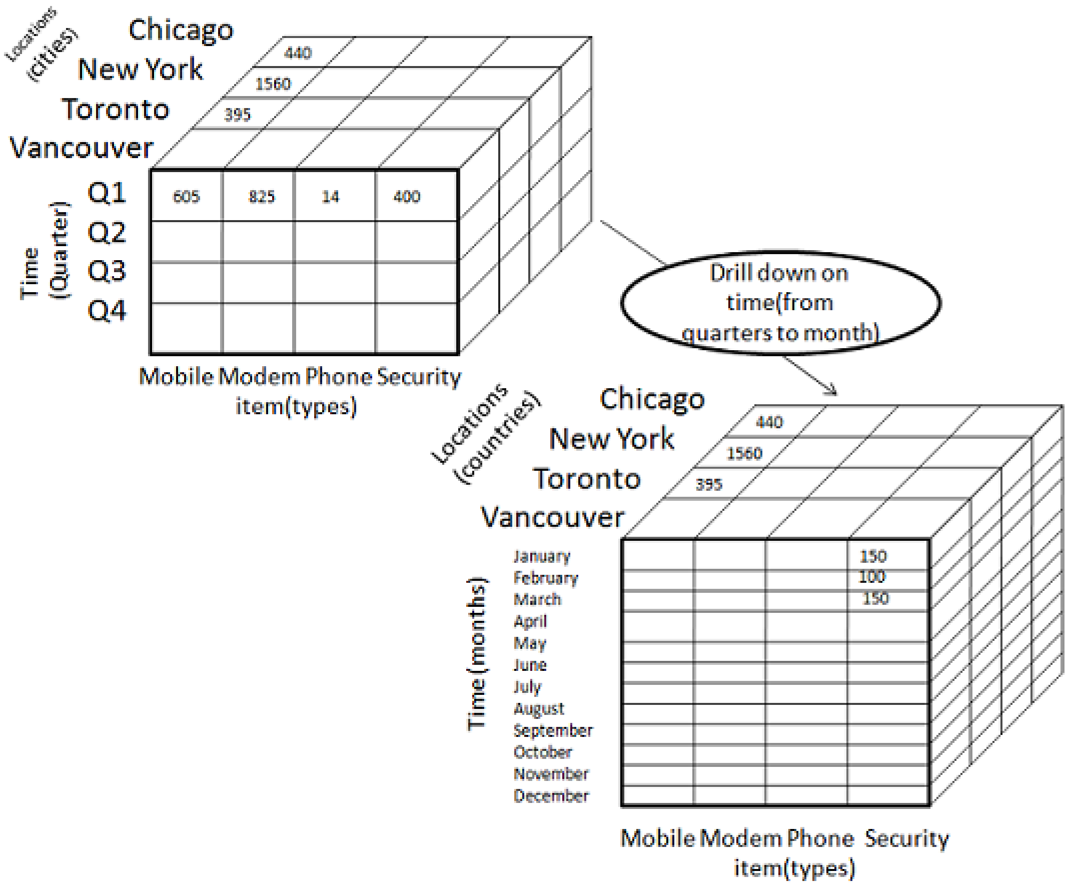
- Roll-up is performed by climbing up a concept hierarchy for the dimension location.
- Initially the concept hierarchy was "street < city < province < country".
- On rolling up, the data is aggregated by ascending the location hierarchy from the level of city to the level of country.
- The data is grouped 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 −

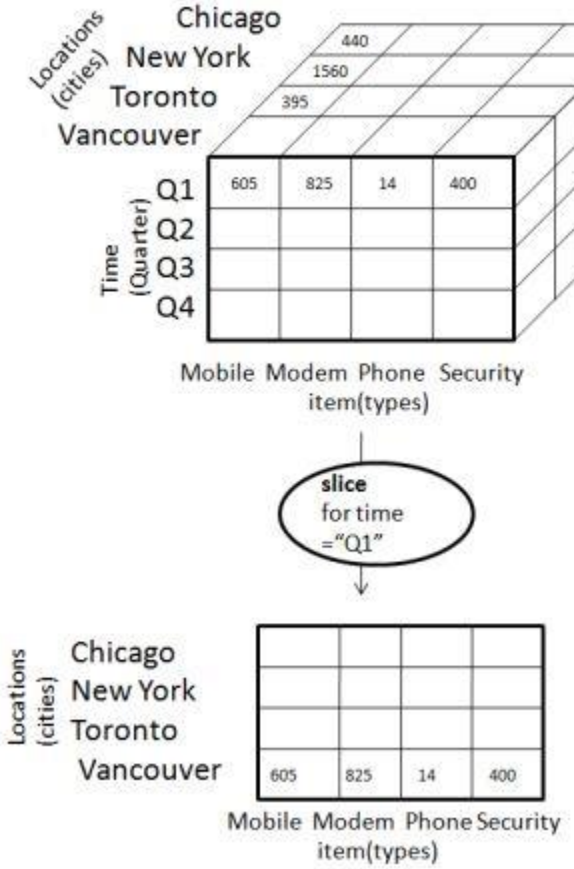


- Drill-down is performed by stepping down a concept hierarchy for the dimension time.

- 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

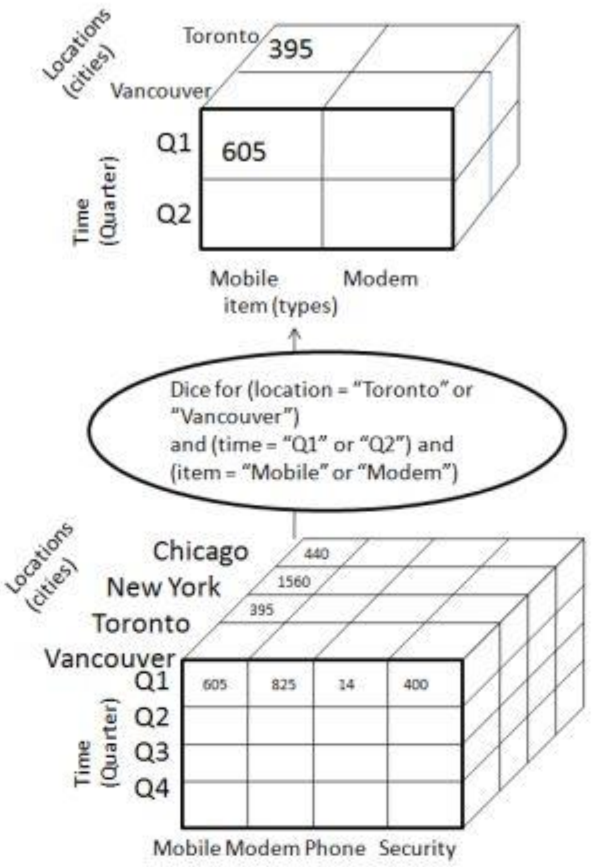
The slice operation selects one particular dimension from a given cube and provides a new sub-cube. Consider the following diagram that shows how slice works.



- 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.

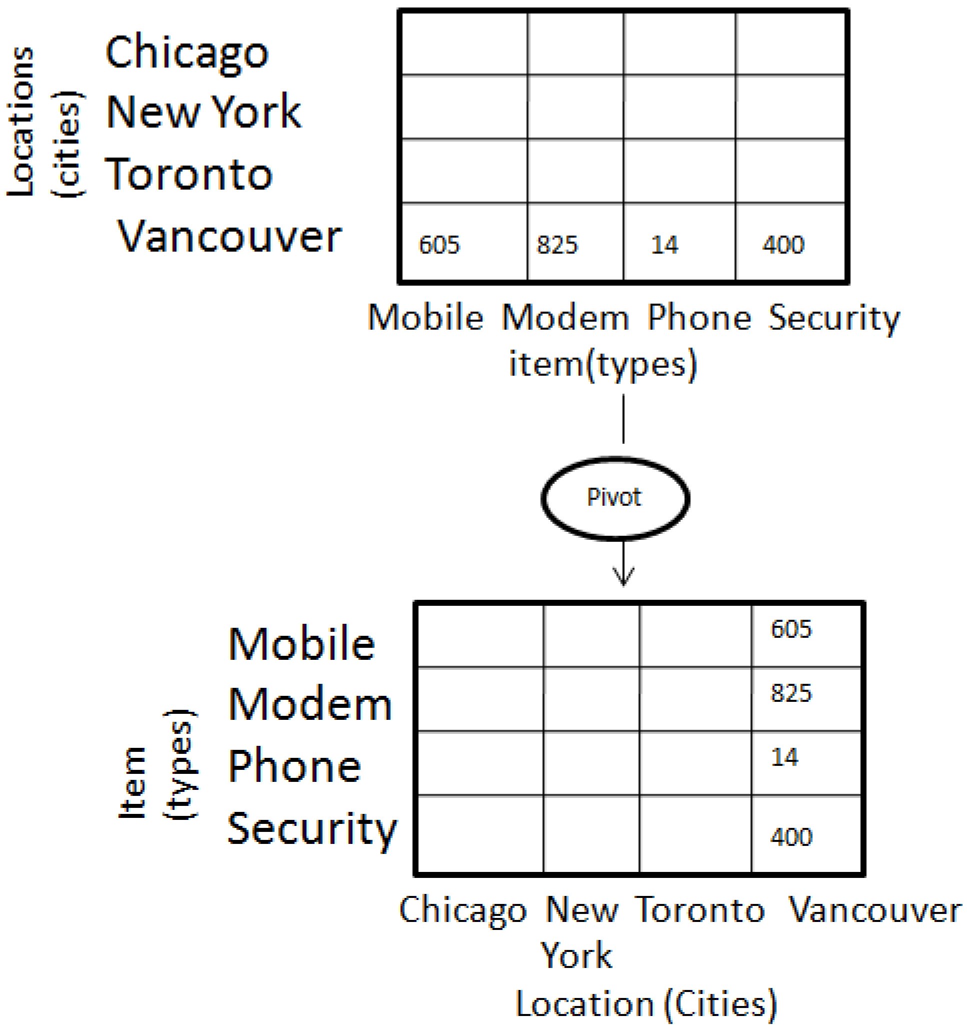


The dice operation on the cube based on the following selection criteria involves three dimensions.

- (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.



	Mobile	Modem	Phone	Security
Chicago				
New York				
Toronto				
Vancouver	605	825	14	400

	Chicago	New York	Toronto	Vancouver
Mobile				605
Modem				825
Phone				14
Security				400

Data Warehouse Modeling

Data warehouse modeling is the process of designing the schemas of the detailed and summarized information of the data warehouse. The goal of data warehouse modeling is to develop a schema describing the reality, or at least a part of the fact, which the data warehouse is needed to support. Data warehouse modeling is an essential stage of building a data warehouse for two main reasons. Firstly, through the schema, data warehouse clients can visualize the relationships among the warehouse data, to use them with greater ease. Secondly, a well-designed schema allows an effective data warehouse structure to emerge, to help decrease the cost of implementing the warehouse and improve the efficiency of using it.

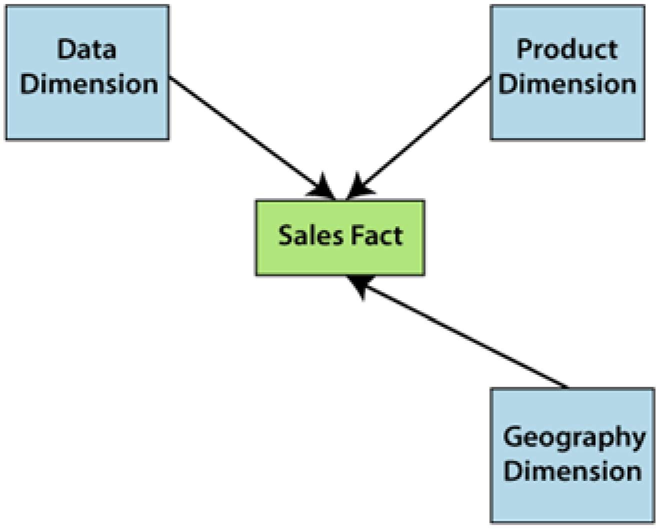
Conceptual Data Model

A conceptual data model recognizes the highest-level relationships between the different entities.

Characteristics of the conceptual data model

- It contains the essential entities and the relationships among them.
- No attribute is specified.
- No primary key is specified.

We can see that the only data shown via the conceptual data model is the entities that define the data and the relationships between those entities. No other data, as shown through the conceptual data model.



Example of Conceptual Data Model

Logical Data Model

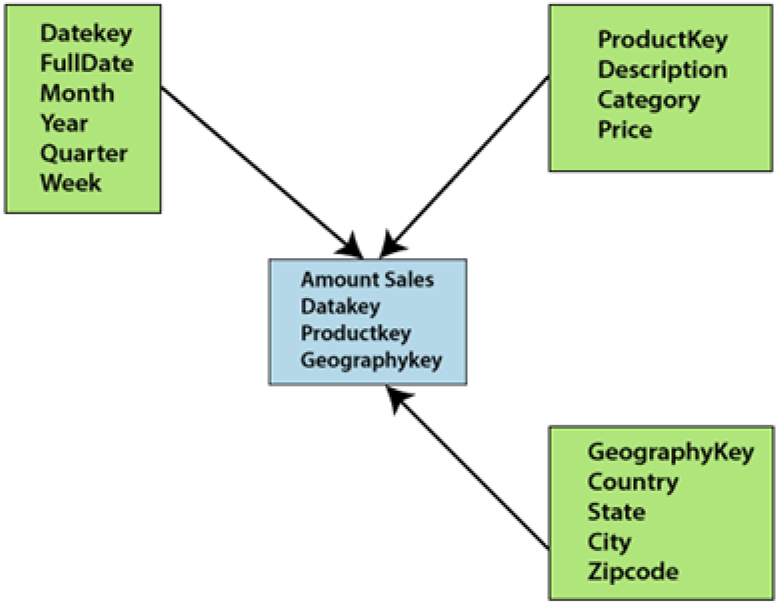
A logical data model defines the information in as much structure as possible, without observing how they will be physically achieved in the database. The primary objective of logical data modeling is to document the business data structures, processes, rules, and relationships by a single view - the logical data model.

Features of a logical data model

- It involves all entities and relationships among them.
- All attributes for each entity are specified.
- The primary key for each entity is stated.
- Referential Integrity is specified (FK Relation).

The phase for designing the logical data model which are as follows:

- Specify primary keys for all entities.
- List the relationships between different entities.
- List all attributes for each entity.
- Normalization.
- No data types are listed



Example of Logical Data Model

Physical Data Model

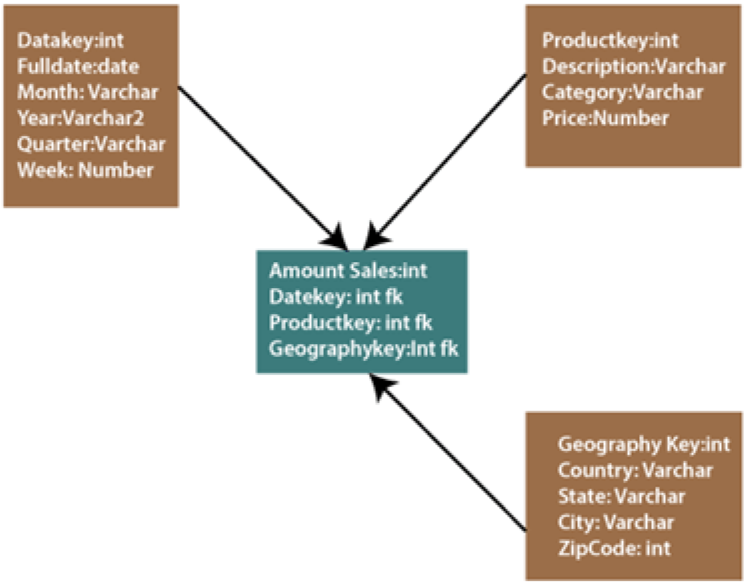
Physical data model describes how the model will be presented in the database. A physical database model demonstrates all table structures, column names, data types, constraints, primary key, foreign key, and relationships between tables. The purpose of physical data modeling is the mapping of the logical data model to the physical structures of the RDBMS system hosting the data warehouse. This contains defining physical RDBMS structures, such as tables and data types to use when storing the information. It may also include the definition of new data structures for enhancing query performance.

Characteristics of a physical data model

- Specification all tables and columns.
- Foreign keys are used to recognize relationships between tables.

The steps for physical data model design which are as follows:

- Convert entities to tables.
- Convert relationships to foreign keys.
- Convert attributes to columns.



Example of Physical Data Model

Data Warehouse Architecture

Data Warehouse Architecture is complex as it's an information system that contains historical and commutative data from multiple sources. There are 3 approaches for constructing Data Warehouse layers: Single Tier, Two tier and Three tier. This 3-tier architecture of Data Warehouse is explained as below.

Single-tier architecture

The objective of a single layer is to minimize the amount of data stored. This goal is to remove data redundancy. This architecture is not frequently used in practice.

Two-tier architecture

Two-layer architecture is one of the Data Warehouse layers which separates physically available sources and data warehouse. This architecture is not expandable and also not supporting a large number of end-users. It also has connectivity problems because of network limitations.

Three-Tier Data Warehouse Architecture

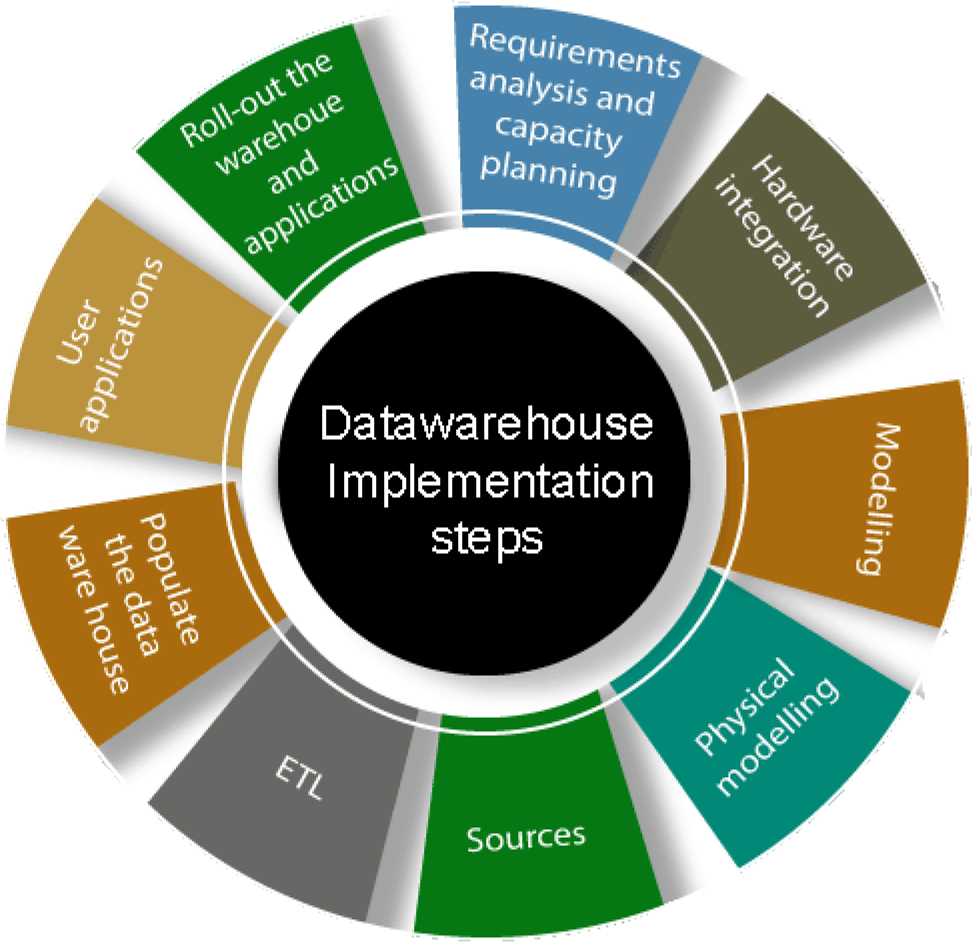
This is the most widely used Architecture of Data Warehouse.

It consists of the Top, Middle and Bottom Tier.

1. **Bottom Tier:** The database of the Datawarehouse servers as the bottom tier. It is usually a relational database system. Data is cleansed, transformed, and loaded into this layer using back-end tools.
2. **Middle Tier:** The middle tier in Data warehouse is an OLAP server which is implemented using either ROLAP or MOLAP model. For a user, this application tier presents an abstracted view of the database. This layer also acts as a mediator between the end-user and the database.
3. **Top-Tier:** The top tier is a front-end client layer. Top tier is the tools and API that you connect and get data out from the data warehouse. It could be Query tools, reporting tools, managed query tools, Analysis tools and Data mining tools.

Data Warehouse Implementation

There are various implementation in data warehouses which are as follows


Datawarehouse Implementation steps
Requirements analysis and capacity planning
Hardware integration
Modelling
Physical modelling
Sources
ETL
Populate the data ware house
User applications
Roll-out the warehoue and applications

1. Requirements analysis and capacity planning: The first process in data warehousing involves defining enterprise needs, defining architectures, carrying out capacity planning, and selecting the hardware and software tools. This step will contain be consulting senior management as well as the different stakeholder.

2. Hardware integration: Once the hardware and software has been selected, they require to be put by integrating the servers, the storage methods, and the user software tools.

3. Modeling: Modelling is a significant stage that involves designing the warehouse schema and views. This may contain using a modeling tool if the data warehouses are sophisticated.

4. Physical modeling: For the data warehouses to perform efficiently, physical modeling is needed. This contains designing the physical data warehouse organization, data placement, data partitioning, deciding on access techniques, and indexing.

5. Sources: The information for the data warehouse is likely to come from several data sources. This step contains identifying and connecting the sources using the gateway, ODBC drives, or another wrapper.

6. ETL: The data from the source system will require to go through an ETL phase. The process of designing and implementing the ETL phase may contain defining a suitable ETL tool vendors and purchasing and implementing the tools. This may contains customize the tool to suit the need of the enterprises.

7. Populate the data warehouses: Once the ETL tools have been agreed upon, testing the tools will be needed, perhaps using a staging area. Once everything is working adequately, the ETL tools may be used in populating the warehouses given the schema and view definition.

8. User applications: For the data warehouses to be helpful, there must be end-user applications. This step contains designing and implementing applications required by the end-users.

9. Roll-out the warehouses and applications: Once the data warehouse has been populated and the end-client applications tested, the warehouse system and the operations may be rolled out for the user's community to use.

What is Data Mart?

A **Data Mart** is focused on a single functional area of an organization and contains a subset of data stored in a Data Warehouse. A Data Mart is a condensed version of Data Warehouse and is designed for use by a specific department, unit or set of users in an organization. E.g., Marketing, Sales, HR or finance. It is often controlled by a single department in an organization.

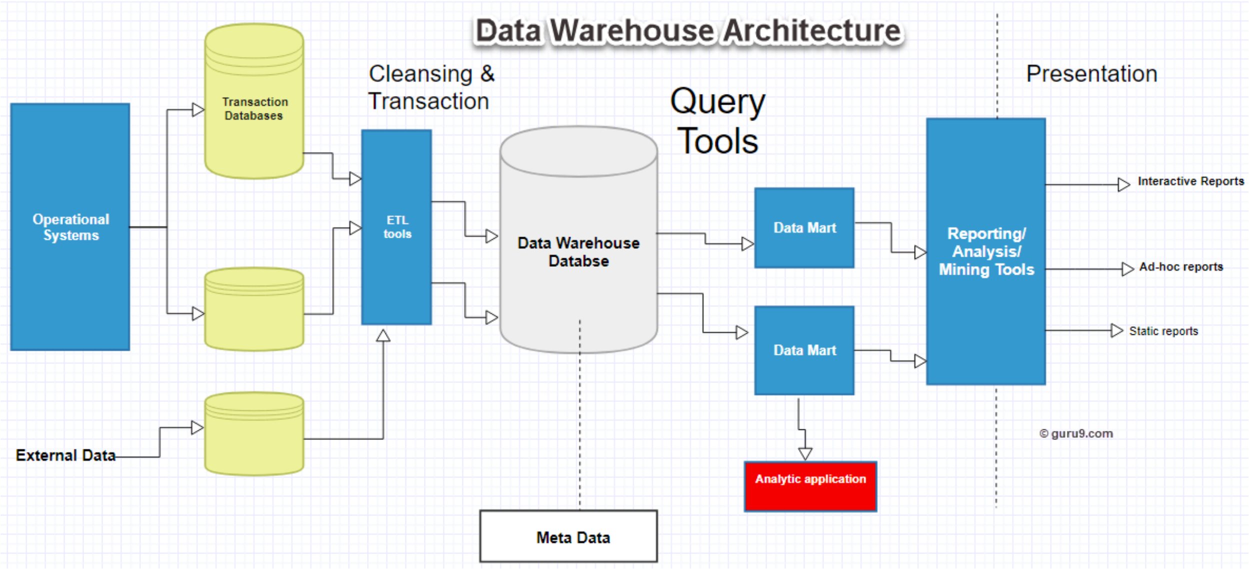
Why do we need Data Mart?

- Data Mart helps to enhance user's response time due to reduction in volume of data
- It provides easy access to frequently requested data.
- Data mart are simpler to implement when compared to corporate Datawarehouse. At the same time, the cost of implementing Data Mart is certainly lower compared with implementing a full data warehouse.
- Compared to Data Warehouse, a datamart is agile. In case of change in model, datamart can be built quicker due to a smaller size.

- A Datamart is defined by a single Subject Matter Expert. On the contrary data warehouse is defined by interdisciplinary SME from a variety of domains. Hence, Data mart is more open to change compared to Datawarehouse.
- Data is partitioned and allows very granular access control privileges.
- Data can be segmented and stored on different hardware/software platforms.

Datawarehouse Components

We will learn about the Datawarehouse Components and Architecture of Data Warehouse with Diagram as shown below:



Data Warehouse Architecture

The Data Warehouse is based on an RDBMS server which is a central information repository that is surrounded by some key Data Warehousing components to make the entire environment functional, manageable and accessible.

There are mainly five Data Warehouse Components:

Data Warehouse Database

The central database is the foundation of the data warehousing environment. This database is implemented on the RDBMS technology. Although, this kind of implementation is constrained by the fact that traditional RDBMS system is optimized for transactional database processing and not for data warehousing. For instance, ad-hoc query, multi-table joins, aggregates are resource intensive and slow down performance.

The central component of a data warehousing architecture is a databank that stocks all enterprise data and makes it manageable for reporting. Obviously, this means you need to choose which kind of database you'll use to store data in your warehouse.

The following are the four database types that you can use:

Typical relational databases are the row-centered databases you perhaps use on an everyday basis. For example, Microsoft SQL Server, SAP, Oracle, and IBM DB2.

Analytics databases are precisely developed for data storage to sustain and manage analytics—for example, Teradata and Greenplum.

Data warehouse applications aren't exactly a kind of storage database, but several dealers now offer applications that offer software for data management as well as hardware for storing data. For example, SAP Hana, Oracle Exadata, and IBM Netezza.

Cloud-based databases can be hosted and retrieved on the cloud so that you don't have to procure any hardware to set up your data warehouse—for example, Amazon Redshift, Google BigQuery, and Microsoft Azure SQL.

Sourcing, Acquisition, Clean-up and Transformation Tools (ETL)

The data sourcing, transformation, and migration tools are used for performing all the conversions, summarizations, and all the changes needed to transform data into a unified format in the Datawarehouse. They are also called Extract, Transform and Load (ETL) Tools.

The ETL tool you choose will determine:

- The time expended in data extraction
- Approaches to extracting data
- Kind of transformations applied and the simplicity to do so
- Business rule definition for data validation and cleansing to improve end-product analytics
- Filling mislaid data

- Outlining information distribution from the fundamental depository to your BI applications
- .
- De-duplicated repeated data arriving from multiple data sources.

These Extract, Transform, and Load tools may generate cron jobs, background jobs, Cobol programs, shell scripts, etc. that regularly update data in Datawarehouse. These tools are also helpful to maintain the Metadata.

These ETL Tools have to deal with challenges of Database & Data heterogeneity.

Metadata

The name Meta Data suggests some high-level technological Data Warehousing Concepts. However, it is quite simple. Metadata is data about data which defines the data warehouse. It is used for building, maintaining and managing the data warehouse.

In the Data Warehouse Architecture, meta-data plays an important role as it specifies the source, usage, values, and features of data warehouse data. It also defines how data can be changed and processed. It is closely connected to the data warehouse.

For example, a line in sales database may contain:

```
4030 KJ732 299.90
```

This is a meaningless data until we consult the Meta that tell us it was

- Model number: 4030
- Sales Agent ID: KJ732
- Total sales amount of $299.90

Therefore, Meta Data are essential ingredients in the transformation of data into knowledge.

Metadata can be classified into following categories:

1. **Technical Meta Data**: This kind of Metadata contains information about warehouse which is used by Data warehouse designers and administrators.

2. **Business Meta Data:** This kind of Metadata contains detail that gives end-users a way easy to understand information stored in the data warehouse.

Query Tools

One of the primary objects of data warehousing is to provide information to businesses to make strategic decisions. Query tools allow users to interact with the data warehouse system.

These tools fall into four different categories:

1. Query and reporting tools
2. Application Development tools
3. Data mining tools
4. OLAP tools

1. Query and reporting tools:

Query and reporting tools can be further divided into

- Reporting tools
- Managed query tools

Reporting tools:

Reporting tools can be further divided into production reporting tools and desktop report writer.

1. Report writers: This kind of reporting tool are tools designed for end-users for their analysis.
2. Production reporting: This kind of tools allows organizations to generate regular operational reports. It also supports high volume batch jobs like printing and calculating. Some popular reporting tools are Brio, Business Objects, Oracle, PowerSoft, SAS Institute.

Managed query tools:

This kind of access tools helps end users to resolve snags in database and SQL and database structure by inserting meta-layer between users and database.

2. Application development tools:

Sometimes built-in graphical and analytical tools do not satisfy the analytical needs of an organization. In such cases, custom reports are developed using Application development tools.

3. Data mining tools:

Data mining is a process of discovering meaningful new correlation, pattens, and trends by mining large amount data. Data mining tools are used to make this process automatic.

4. OLAP tools:

These tools are based on concepts of a multidimensional database. It allows users to analyse the data using elaborate and complex multidimensional views.

Data warehouse Bus Architecture

Data warehouse Bus determines the flow of data in your warehouse. The data flow in a data warehouse can be categorized as Inflow, Upflow, Downflow, Outflow and Meta flow.

While designing a Data Bus, one needs to consider the shared dimensions, facts across data marts.

Data Marts

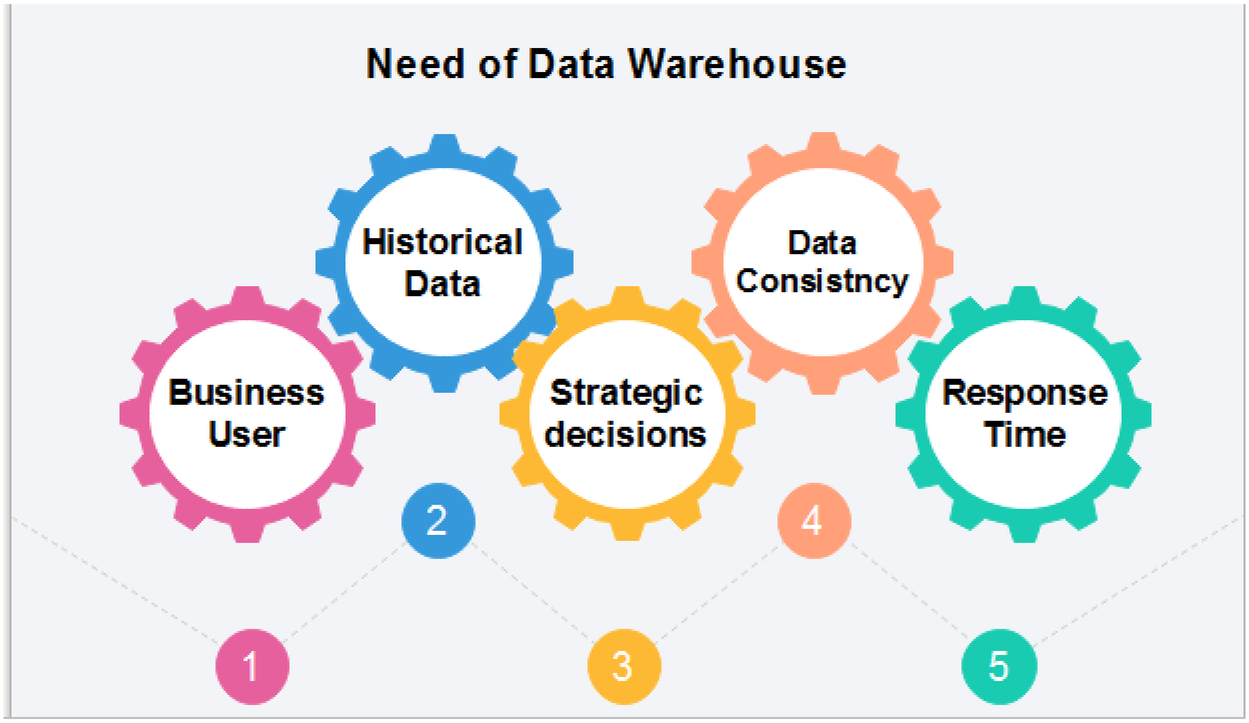
A data mart is an access layer which is used to get data out to the users. It is presented as an option for large size data warehouse as it takes less time and money to build. However, there is no standard definition of a data mart is differing from person to person.

In a simple word Data mart is a subsidiary of a data warehouse. The data mart is used for partition of data which is created for the specific group of users.

Data marts could be created in the same database as the Datawarehouse or a physically separate Database.

Need for Data Warehouse

Data Warehouse is needed for the following reasons:



1. 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. 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. 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. 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. 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

ADVERTISEMENT

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.

Disadvantages of Data Warehouse:

- Not an ideal option for unstructured data.
- Creation and Implementation of Data Warehouse is surely time confusing affair.
- Data Warehouse can be outdated relatively quickly
- Difficult to make changes in data types and ranges, data source schema, indexes, and queries.
- The data warehouse may seem easy, but actually, it is too complex for the average users.
- Despite best efforts at project management, data warehousing project scope will always increase.
- Sometime warehouse users will develop different business rules.
- Organisations need to spend lots of their resources for training and Implementation purpose.

Characteristics and Functions of Data warehouse

1. **Subject-oriented –**
   A data warehouse is always a subject oriented as it delivers information about a theme instead of organization's current operations. It can be achieved on specific theme. That means the data warehousing process is proposed to handle with a specific theme

which is more defined. These themes can be sales, distributions, marketing etc.

A data warehouse never put emphasis only current operations. Instead, it focuses on demonstrating and analysis of data to make various decision. It also delivers an easy and precise demonstration around particular theme by eliminating data which is not required to make the decisions.

2. **Integrated –**
   It is somewhere same as subject orientation which is made in a reliable format. Integration means founding a shared entity to scale the all similar data from the different databases. The data also required to be resided into various data warehouse in shared and generally granted manner.

   A data warehouse is built by integrating data from various sources of data such that a mainframe and a relational database. In addition, it must have reliable naming conventions, format and codes. Integration of data warehouse benefits in effective analysis of data. Reliability in naming conventions, column scaling, encoding structure etc. should be confirmed. Integration of data warehouse handles various subject related warehouse.
3. **Time-Variant –**
   In this data is maintained via different intervals of time such as weekly, monthly, or annually etc. It founds various time limit which are structured between the large datasets and are held in online transaction process (OLTP). The time limits for data warehouse is wide-ranged than that of operational systems. The data resided in data warehouse is predictable with a specific interval of time and delivers information from the historical perspective. It comprises elements of time explicitly or implicitly. Another feature of time-variance is that once data is stored in the data warehouse then it cannot be modified, alter, or updated.
4. **Non-Volatile –**
   As the name defines the data resided in data warehouse is permanent. It also means that data is not erased or deleted when new data is inserted. It includes the mammoth quantity of data that is inserted into modification between the selected quantity on logical business. It evaluates the analysis within the technologies of warehouse.

   In this, data is read-only and refreshed at particular intervals. This is beneficial in analysing historical data and in comprehension the functionality. It does not need transaction process, recapture and concurrency control mechanism. Functionalities such as delete, update, and insert that are done in an operational application are lost in data warehouse environment. Two types of data operations done in the data warehouse are:

- Data Loading
- Data Access

Functions of Data warehouse:

It works as a collection of data and here is organized by various communities that endures the features to recover the data functions. It has stocked facts about the tables which have high transaction levels which are observed so as to define the data warehousing techniques and major functions which are involved in this are mentioned below:

1. Data consolidation
2. Data Cleaning
3. Data Integration

Trends in data warehousing

1. Complex Data Marts Will Define the Future Business Models

Data marts surfaced as a subset of data warehouses, designed to address the requirements of a specific business function. However, the ability of large and complex data marts to pull data from disparate sources and make it accessible to business users is making it a rising trend in data warehousing.

2. Column-based Storage is on the Rise

When it comes to retrieving analytical queries, the efficiency of column-based storage is higher than its row-based alternative. This is one of the reasons this trend is gradually gaining popularity.

The primary goal of data warehousing is to store data in a way that speeds up the query response time, consequently enabling efficient data evaluation and analyzation.

3. Mixed Workloads Are Becoming Common

A data warehouse platform delivers six types of workloads:

- Basic reporting
- Continuous/real-time load
- Batch/bulk load
- Operational BI
- Online analytical processing (OLAP)
- Data mining

4. Data Warehouse Automation (DWA)

Data warehouse implementations are generally dependent on IT personnel. It can take years to build a data warehouse, making the whole process time-intensive, expensive, and slow. Adding the automation factor to the equation makes it easier for organizations to navigate the complexities of data warehousing and eliminates the repetitive, time-consuming tasks from the process cycle. This consequently results in low project costs and high productivity.

5. Data Warehouses are Becoming Cloud-centric

The cloud is fast becoming a preferred choice for users looking to acquire data warehousing capabilities. Why? Because in addition to supporting all the functions like that of a traditional data warehouse, cloud data warehouses optimize deployments like data-governance hubs, BI backends, analytic data marts, etc.

Motivation and Background

A host of technological advances have resulted in generating a huge amount of electronic data, and have enabled the data to be captured, processed, analyzed, and stored rather inexpensively. This capability has enabled industries and innovations such as

- Banking, insurance, financial transactions - electronic banking, ATMs, credit cards, stock market data
- Supermarket check-out scanner data, point-of-sale devices, barcode readers
- Healthcare - pharmaceutical records
- Communications - telephone-call detail records
- Location data - GPS, cell phones
- Internet and e-commerce - Web logs, click-streams that generate huge volumes of electronic data. For example, Walmart has 20 million transactions/day and a 10 terabyte database, and Blockbuster has over 36 million household customers.

The need to understand huge, complex, information-rich data sets is important to virtually all fields in business, science and engineering. The ability to extract useful knowledge hidden in these data and to act on that knowledge is becoming vital in today's increasingly competitive world. Such data (typically terabytes in size) is often stored in data warehouses and data marts.

Introduction to data mining system

Data Mining, also popularly known as *Knowledge Discovery in Databases* (KDD), refers to the nontrivial extraction of implicit, previously unknown and potentially useful information from data in databases. While data mining and knowledge discovery in databases (or KDD) are frequently treated as synonyms, data mining is actually part of the knowledge discovery process.

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. For example. To study the characteristics of a software product whose sales increased by

15% two years ago, anyone can collect these type of data related to such products by running SQL queries.

- **Data Discrimination:**
  It compares common features of class which is under study. The output of this process can be represented in many forms. Eg., bar charts, curves and pie charts.

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 frequency 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. for example, it can be used to determine the sales of items that are frequently purchased together.

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, Heighted people tend to have more weight.

Knowledge Discovery in Databases or KDD

Knowledge discovery as a process is depicted and consists of an iterative sequence of the following steps:

- Data cleaning (to remove noise or irrelevant data),
- Data integration (where multiple data sources may be combined)
- Data selection (where data relevant to the analysis task are retrieved from the database)
- Data transformation (where data are transformed or consolidated into forms appropriate for mining by performing summary or aggregation operations, for instance),
- Data mining (an essential process where intelligent methods are applied in order to extract data patterns),

- Pattern evaluation (to identify the truly interesting patterns representing knowledge based on some interestingness measures;), and

- Knowledge presentation (where visualization and knowledge representation techniques are used to present the mined knowledge to the user

Data

- Collection of data objects and their attributes

- An attribute is a property or characteristic of an object

    – Examples: eye color of a person, temperature, etc.

    – Attribute is also known as variable, field, characteristic, or feature

- A collection of attributes describe an object

    – Object is also known as record, point, case, sample, entity, or instance

    Attributes

Attribute Values

- Attribute values are numbers or symbols assigned to an attribute

    - Distinction between attributes and attribute values

– Same attribute can be mapped to different attribute values

    - Example: height can be measured in feet or meters

– Different attributes can be mapped to the same set of values

- Example: Attribute values for ID and age are integers

- But properties of attribute values can be different

    – ID has no limit but age has a maximum and minimum value

Types of Attributes

- There are different types of attributes

– Nominal

- Examples: ID numbers, eye color, zip codes

– Ordinal

- Examples: rankings (e.g., taste of potato chips on a scale from 1-10),

grades, height in {tall, medium, short}

– Interval

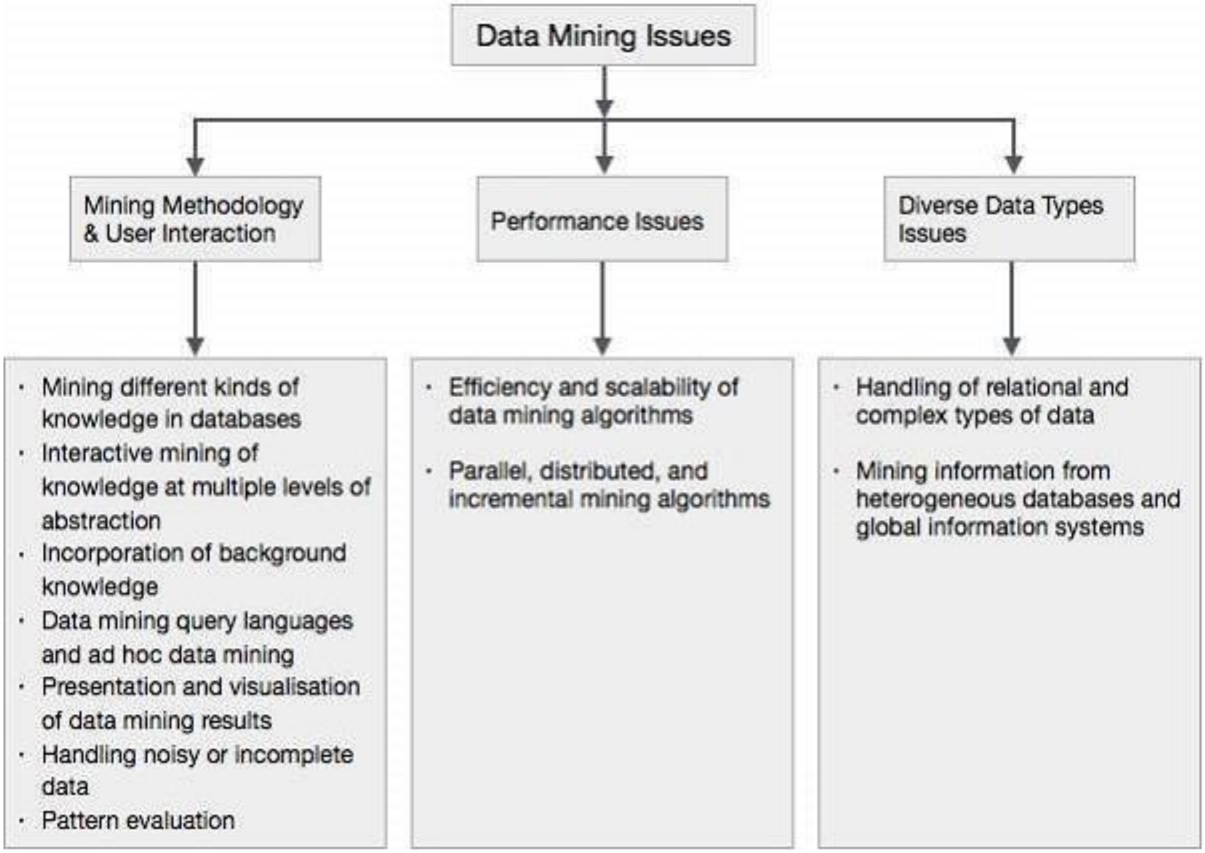
- Examples: calendar dates, temperatures in Celsius or Fahrenheit.

– Ratio

Examples: temperature in Kelvin, length, time, counts

Issues and Applications

The following diagram describes the major issues.


Data Mining Issues
Mining Methodology & User Interaction
Performance Issues
Diverse Data Types Issues
Mining different kinds of knowledge in databases
Interactive mining of knowledge at multiple levels of abstraction
Incorporation of background knowledge
Data mining query languages and ad hoc data mining
Presentation and visualisation of data mining results
Handling noisy or incomplete data
Pattern evaluation
Efficiency and scalability of data mining algorithms
Parallel, distributed, and incremental mining algorithms
Handling of relational and complex types of data
Mining information from heterogeneous databases and global information systems

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 Mining Applications

Here is the list of areas where data mining is widely used −

- Financial Data Analysis
- Retail Industry
- Telecommunication Industry
- Biological Data Analysis
- Other Scientific Applications
- Intrusion Detection

Financial Data Analysis

The financial data in banking and financial industry is generally reliable and of high quality which facilitates systematic data analysis and data mining. Some of the typical cases are as follows −

- Design and construction of data warehouses for multidimensional data analysis and data mining.
- Loan payment prediction and customer credit policy analysis.
- Classification and clustering of customers for targeted marketing.
- Detection of money laundering and other financial crimes.

Retail Industry

Data Mining has its great application in Retail Industry because it collects large amount of data from on sales, customer purchasing history, goods transportation, consumption and services. It is natural that the quantity of data collected will continue to expand rapidly because of the increasing ease, availability and popularity of the web.

Data mining in retail industry helps in identifying customer buying patterns and trends that lead to improved quality of customer service and good customer retention and satisfaction. Here is the list of examples of data mining in the retail industry −

- Design and Construction of data warehouses based on the benefits of data mining.
- Multidimensional analysis of sales, customers, products, time and region.

- Analysis of effectiveness of sales campaigns.
- Customer Retention.
- Product recommendation and cross-referencing of items.

Telecommunication Industry

Today the telecommunication industry is one of the most emerging industries providing various services such as fax, pager, cellular phone, internet messenger, images, e-mail, web data transmission, etc. Due to the development of new computer and communication technologies, the telecommunication industry is rapidly expanding. This is the reason why data mining is become very important to help and understand the business.

Data mining in telecommunication industry helps in identifying the telecommunication patterns, catch fraudulent activities, make better use of resource, and improve quality of service. Here is the list of examples for which data mining improves telecommunication services −

- Multidimensional Analysis of Telecommunication data.
- Fraudulent pattern analysis.
- Identification of unusual patterns.
- Multidimensional association and sequential patterns analysis.
- Mobile Telecommunication services.
- Use of visualization tools in telecommunication data analysis.

Biological Data Analysis

In recent times, we have seen a tremendous growth in the field of biology such as genomics, proteomics, functional Genomics and biomedical research. Biological data mining is a very important part of Bioinformatics. Following are the aspects in which data mining contributes for biological data analysis −

- Semantic integration of heterogeneous, distributed genomic and proteomic databases.
- Alignment, indexing, similarity search and comparative analysis multiple nucleotide sequences.
- Discovery of structural patterns and analysis of genetic networks and protein pathways.
- Association and path analysis.
- Visualization tools in genetic data analysis.

Other Scientific Applications

The applications discussed above tend to handle relatively small and homogeneous data sets for which the statistical techniques are appropriate. Huge amount of data have been collected from scientific domains such as geosciences, astronomy, etc. A large amount of data sets is being generated because of the fast numerical simulations in various fields such as climate and ecosystem modeling, chemical engineering, fluid dynamics, etc. Following are the applications of data mining in the field of Scientific Applications −

- Data Warehouses and data preprocessing.
- Graph-based mining.
- Visualization and domain specific knowledge.

Intrusion Detection

Intrusion refers to any kind of action that threatens integrity, confidentiality, or the availability of network resources. In this world of connectivity, security has become the major issue. With increased usage of internet and availability of the tools and tricks for intruding and attacking network prompted intrusion detection to become a critical component of network administration. Here is the list of areas in which data mining technology may be applied for intrusion detection −

- Development of data mining algorithm for intrusion detection.
- Association and correlation analysis, aggregation to help select and build discriminating attributes.
- Analysis of Stream data.
- Distributed data mining.
- Visualization and query tools.