Introduction to Data Warehousing

What is Data Warehousing?

A data warehousing is defined as a technique for collecting and managing data from varied sources to provide meaningful business insights. It is a blend of technologies and components which aids the strategic use of data.

It is electronic storage of a large amount of information by a business which is designed for query and analysis instead of transaction processing. It is a process of transforming data into information and making it available to users in a timely manner to make a difference.

Difference between Database and Data Warehouse

Parameter	Database	Data Warehouse
Purpose	Is designed to record	Is designed to analyze
Processing Method	The database uses the Online Transactional Processing (OLTP)	Data warehouse uses Online Analytical Processing (OLAP).
Usage	The database helps to perform fundamental operations for your business	Data warehouse allows you to analyze your business.
Tables and Joins	Tables and joins of a database are complex as they are normalized.	Table and joins are simple in a data warehouse because they are denormalized.

Orientation	Is an application-oriented collection of data	It is a subject-oriented collection of data
Storage limit	Generally limited to a single application	Stores data from any number of applications
Availability	Data is available real-time	Data is refreshed from source systems as and when needed
Usage	ER modeling techniques are used for designing.	Data modeling techniques are used for designing.
Technique	Capture data	Analyze data
Data Type	Data stored in the Database is up to date.	Current and Historical Data is stored in Data Warehouse. May not be up to date.
Storage of data	Flat Relational Approach method is used for data storage.	Data Ware House uses dimensional and normalized approach for the data structure. Example: Star and snowflake schema.
Query Type	Simple transaction queries are used.	Complex queries are used for analysis purpose.
Data Summary	Detailed Data is stored in a database.	It stores highly summarized data.

The compelling need for Data Warehousing

- Companies are desperate for strategic information to counter fiercer competition, extend market share, and improve profitability.
- In spite of tons of data accumulated by enterprises over the past decades, every enterprise is caught in the middle of an information crisis. Information needed for strategic decision making is not readily available.
- All the past attempts by IT to provide strategic information have been failures. This was mainly because IT has been trying to provide strategic information from operational systems.
- Informational systems are different from the traditional operational systems. Operational systems are not designed for strategic information.
- We need a new type of computing environment to provide strategic information. The data warehouse promises to be this new computing environment.
- Data warehousing is the viable solution. There is a compelling need for data warehousing for every enterprise.

Data warehouse – The building Blocks

Defining Features (or Characteristics) and functions of Data warehousing

Data warehouse can be controlled when the user has a shared way of explaining the trends that are introduced as specific subject. Below are major **characteristics** 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

Data Warehouse and Data Mart

What is Data Warehouse?

A Data Warehouse collects and manages data from varied sources to provide meaningful business insights.

It is a collection of data which is separate from the operational systems and supports the decision making of the company. In Data Warehouse data is stored from a historical perspective.

The data in the warehouse is extracted from multiple functional units. It is checked, cleansed and then integrated with Data warehouse system. Data warehouse used a very fast computer system having large storage capacity. This tool can answer any complex queries relating data.

What is Data Mart?

A data mart is a simple form of a Data Warehouse. It is focused on a single subject. Data Mart draws data from only a few sources. These sources may be central Data warehouse, internal operational systems, or external data sources.

A Data Mart is an index and extraction system. It is an important subset of a data warehouse. It is subject-oriented, and it is designed to meet the needs of a specific group of users. Data marts are fast and easy to use, as they make use of small amounts of data.

Differences between Data Warehouse and Data Mart

Parameter	Data Warehouse	Data Mart
Definition	A Data Warehouse is a large repository of data collected from different organizations or departments within a corporation.	A data mart is an only subtype of a Data Warehouse. It is designed to meet the need of a certain user group.
Usage	It helps to take a strategic decision.	It helps to take tactical decisions for the business.
Objective	The main objective of Data Warehouse is to provide an integrated environment and coherent picture of the business at a point in time.	A data mart mostly used in a business division at the department level.
Designing	The designing process of Data Warehouse is quite difficult.	The designing process of Data Mart is easy.
	May or may not use in a dimensional model. However, it can feed dimensional models.	It is built focused on a dimensional model using a start schema.
Data Handling	Data warehousing includes large area of the corporation which is why it takes a long time to process it.	Data marts are easy to use, design and implement as it can only handle small amounts of data.

Focus	Data warehousing is broadly focused all the departments. It is possible that it can even represent the entire company.	Data Mart is subject-oriented, and it is used at a department level.
Data type	The data stored inside the Data Warehouse are always detailed when compared with data mart.	Data Marts are built for particular user groups. Therefore, data short and limited.
Subject-area	The main objective of Data Warehouse is to provide an integrated environment and coherent picture of the business at a point in time.	Mostly hold only one subject area- for example, Sales figure.
Data storing	Designed to store enterprise-wide decision data, not just marketing data.	Dimensional modeling and star schema design employed for optimizing the performance of access layer.
Data type	Time variance and non-volatile design are strictly enforced.	Mostly includes consolidation data structures to meet subject area's query and reporting needs.
Data value	Read-Only from the end-users standpoint.	Transaction data regardless of grain fed directly from the Data Warehouse.

Scope	Data warehousing is more helpful as it can bring information from any department.	Data mart contains data, of a specific department of a company. There are maybe separate data marts for sales, finance, marketing, etc. Has limited usage
Source	In Data Warehouse Data comes from many sources.	In Data Mart data comes from very few sources.
Size	The size of the Data Warehouse may range from 100 GB to 1 TB+.	The Size of Data Mart is less than 100 GB.
Implementation time	The implementation process of Data Warehouse can be extended from months to years.	The implementation process of Data Mart is restricted to few months.

Datawarehouse Components

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

There are 5 main **components** of a **Datawarehouse**. 1) Database 2) ETL Tools 3) Meta **Data** 4) Query Tools 5) DataMarts.

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.

Hence, alternative approaches to Database are used as listed below-

- In a datawarehouse, relational databases are deployed in parallel to allow for scalability. Parallel relational databases also allow shared memory or shared nothing model on various multiprocessor configurations or massively parallel processors.
- New index structures are used to bypass relational table scan and improve speed.
- Use of multidimensional database (MDDBs) to overcome any limitations which are placed because of the relational data model. Example: Essbase from Oracle.

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.

Their functionality includes:

- Anonymize data as per regulatory stipulations.
- Eliminating unwanted data in operational databases from loading into Data warehouse.
- Search and replace common names and definitions for data arriving from different sources.
- Calculating summaries and derived data
- In case of missing data, populate them with defaults.
- De-duplicated repeated data arriving from multiple datasources.

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 concept. 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 helps to answer the following questions

- What tables, attributes, and keys does the Data Warehouse contain?
- Where did the data come from?
- How many times do data get reloaded?
- What transformations were applied with cleansing?

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 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.

Data Warehouse Architectures

There are mainly three types of Datawarehouse Architectures: -

Single-tier architecture

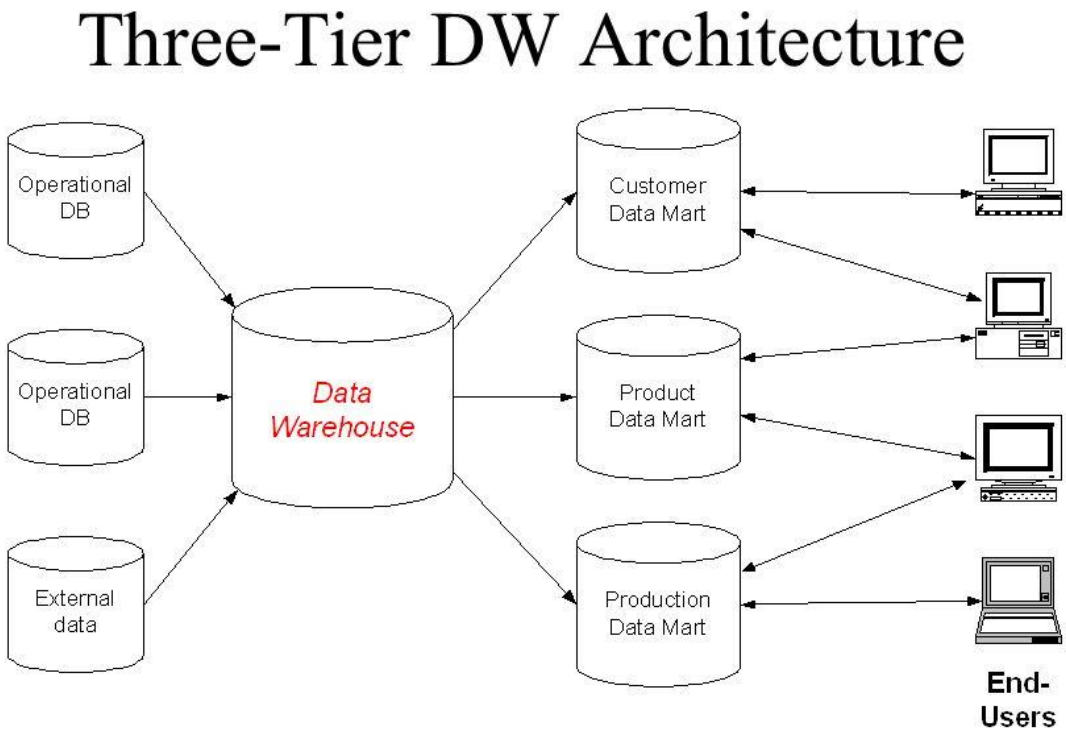
The objective of a single layer is to minimize the amount of data stored. This goal is to remove data redundancy. This architecture is not frequently used in practice.

Two-tier architecture

Two-layer architecture 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 architecture

This is the most widely used architecture.



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 pre-processing

What is data cleaning?

Data cleaning is also known as data scrubbing. Data cleaning is a process which ensures the set of data is correct and accurate. Data accuracy and consistency, data integration is checked during data cleaning. Data cleaning can be applied for a set of records or multiple sets of data which need to be merged.

Data cleaning is performed by reading all records in a set and verifying their accuracy. Typos and spelling errors are rectified. Mislabelled data if available is labelled and filed. Incomplete or missing entries are completed. Unrecoverable records are purged, for not to take space and inefficient operations.

Data cleaning is the process of identifying erroneous data. The data is checked for accuracy, consistency, typos etc.

Methods:-

Parsing - Used to detect syntax errors.
Data Transformation - Confirms that the input data matches in format with expected data.
Duplicate elimination - This process gets rid of duplicate entries.
Statistical Methods- Values of mean, standard deviation, range, or clustering algorithms etc are used to find erroneous data.

Data Transformation

In data transformation process data are transformed from one format to another format, that is more appropriate for data mining.

Some Data Transformation Strategies: -

1 Smoothing

Smoothing is a process of removing noise from the data.

2 Aggregation
Aggregation is a process where summary or aggregation operations are applied to the data.

3 Generalization
In generalization low-level data are replaced with high-level data by using concept hierarchies climbing.

4 Normalization
Normalization scaled attribute data so as to fall within a small specified range, such as 0.0 to 1.0.

5 Attribute Construction
In Attribute construction, new attributes are constructed from the given set of attributes.

ETL (Extract, Transform, and Load) Process

What is ETL?

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

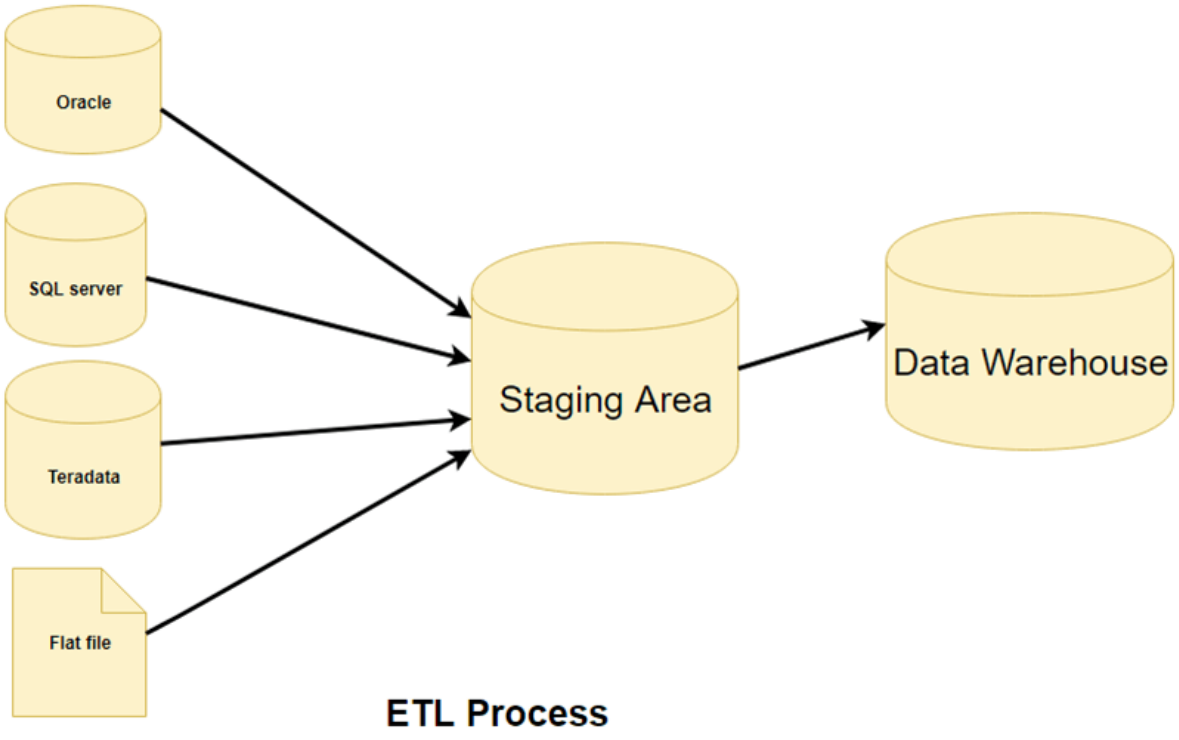
It's tempting to think a creating a Data warehouse is simply extracting data from multiple sources and loading into database of a Data warehouse. This is far from the truth and requires a complex ETL process. The ETL process requires active inputs from various stakeholders including developers, analysts, 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

Step 1) Extraction

In this step, 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 needs 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:

1. Full Extraction
2. Partial Extraction- without update notification.
3. 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 effect company's bottom line.

Some validations are done during Extraction:

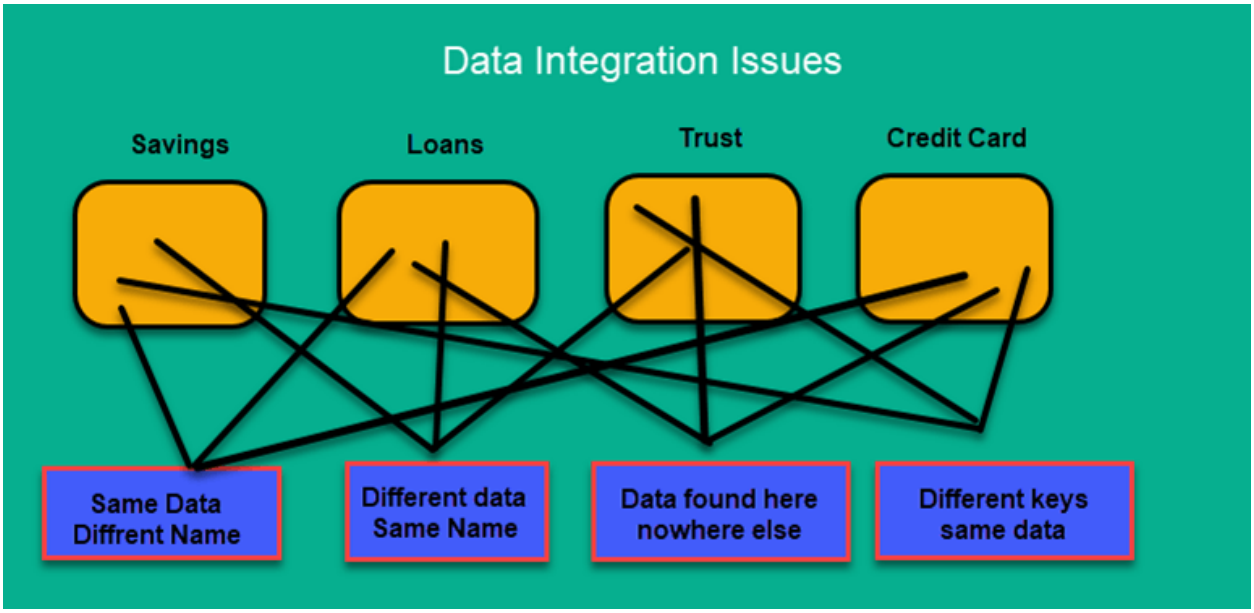
- Reconcile records with the source data
- Make sure that no spam/unwanted data loaded
- Data type check
- Remove all types of duplicate/fragmented data
- Check whether all the keys are in place or not

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.

In this step, 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:

1. Different spelling of the same person like Jon, John, etc.
2. There are multiple ways to denote company name like Google, Google Inc.
3. Use of different names like Cleaveland, Cleveland.
4. There may be a case that different account numbers are generated by various applications for the same customer.
5. In some data required files remains blank
6. 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 like 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 datawarehouse 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.

Best practices ETL process

Never try to cleanse all the data:

Every organization would like to have all the data clean, but most of them are not ready to pay to wait or not ready to wait. To clean it all would simply take too long, so it is better not to try to cleanse all the data.

Never cleanse Anything:

Always plan to clean something because the biggest reason for building the Data Warehouse is to offer cleaner and more reliable data.

Determine the cost of cleansing the data:

Before cleansing all the dirty data, it is important for you to determine the cleansing cost for every dirty data element.

To speed up query processing, have auxiliary views and indexes:

To reduce storage costs, store summarized data into disk tapes. Also, the trade-off between the volume of data to be stored and its detailed usage is required. Trade-off at the level of granularity of data to decrease the storage costs.

ETL Tools

There are many Data Warehousing tools are available in the market. Here, are some most prominent one:

1. MarkLogic:

MarkLogic is a data warehousing solution which makes data integration easier and faster using an array of enterprise features. It can query different types of data like documents, relationships, and metadata.

2. Oracle:

Oracle is the industry-leading database. It offers a wide range of choice of Data Warehouse solutions for both on-premises and in the cloud. It helps to optimize customer experiences by increasing operational efficiency.

3. Amazon RedShift:

Amazon Redshift is Datawarehouse tool. It is a simple and cost-effective tool to analyze all types of data using standard SQL and existing BI tools. It also allows running complex queries against petabytes of structured data.

Defining the business requirements

DIMENSIONAL ANALYSIS

One approach to data warehouse design is to develop and implement a dimensional model. This has given rise to dimensional analysis (sometimes generalized as multi-dimensional analysis).

It was noticed quite early on when data warehouses started to be developed that, whenever decision makers were asked to describe the kinds of questions they would like to get answers to regarding their organizations, they almost always wanted the following:

- Summarized information with the ability to break the summaries into more detail
- Analysis of the summarized information across their own organizational components such as departments or regions
- Ability to slice and dice the information in any way they chose
- Display of the information in both graphical and tabular form
- Capability to view their information over time

So the concept of dimensional analysis became a method for defining data warehouses.

The approach is to determine, by interviewing the appropriate decision makers in an organization, which is the *subject area* that they are most interested in, and which are the most important *dimensions of analysis.*

Recall that one of the characteristics of a data warehouse is that it is subject oriented. The subject area reflects the subject-oriented nature of the warehouse.

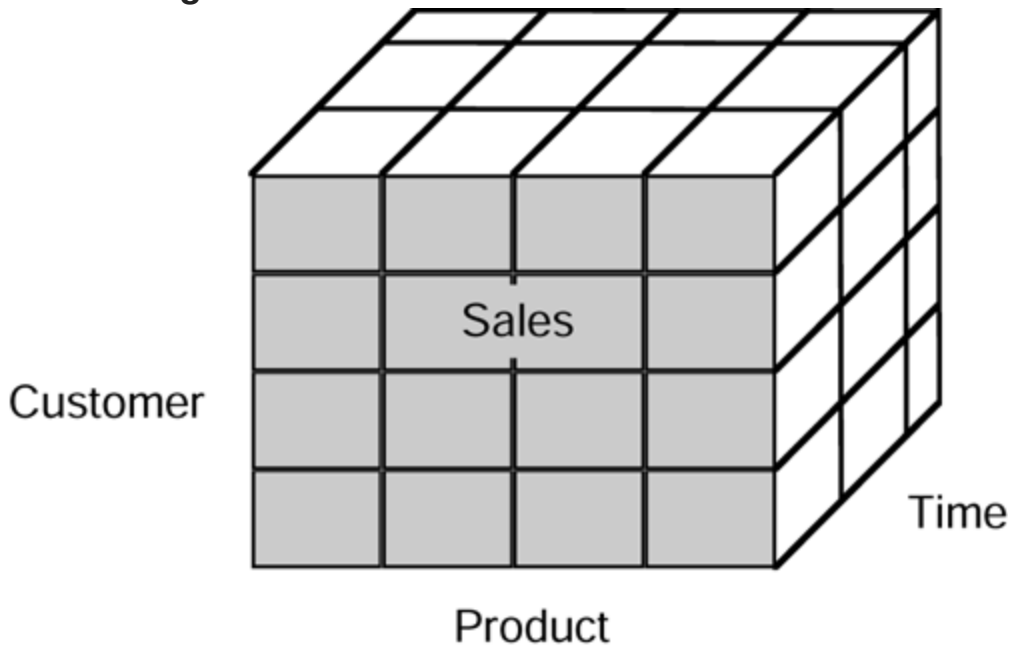
In the example above, the subject area would be Sales. The dimensions of analysis would be Customers and Products. The requirement is to analyze sales by customer and sales by product.

This requirement is depicted in the following three-dimensional cube. Figure 2.2 shows Sales (the shaded area) having axes of:

1. Customer
2. Product

3. Time

Figure 2.2. Three-dimensional data cube.

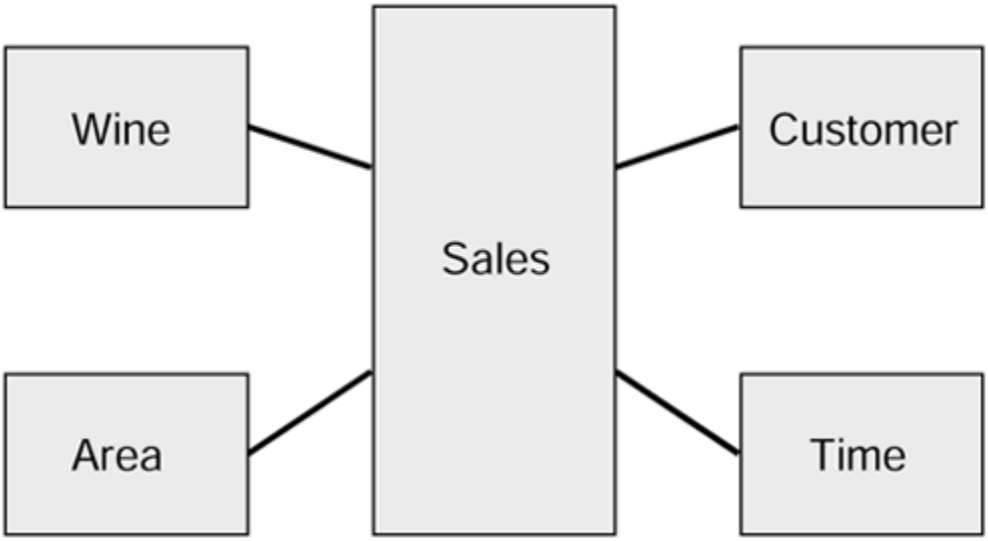


Notice that **time** has not been examined so far. Time is regarded as a necessary dimension of analysis (recall that time variance is another characteristic of data warehouses) and so is always included as one of the dimensions of analysis.

This means that Sales can be analyzed by Customer by Product over Time. So each element of the cube (each minicube) contains a value for sales to a particular customer, of a particular product, at a particular point in time.

As we cannot draw four-dimensional models, we can represent the conceptual dimensional model as shown in Figure 2.3.

Figure 2.3. Wine sales dimensional model for the Wine Club.



The diagram in Figure 2.3 is often referred to as a *Star Schema* because the diagram loosely resembles a star shape. The subject area is the center of the star and the dimensions of analysis form the points of the star. The subject area is often drawn long and thin because the table itself is usually long and thin in that it contains a small number of columns but a very large number of rows.

The Star Schema is the most commonly used diagram for dimensional models.

Information Packages

- Accurate requirements definition in a data warehouse project is many times more important than in other types of projects. Clearly understand the impact of business requirements on every development phase.
- Business requirements condition the outcome of the data design phase.
- Every component of the data warehouse architecture is strongly influenced by the business requirements.

- In order to provide data quality, identify the data pollution sources, the prevalent types of quality problems, and the means to eliminate data corruption early in the requirements definition phase itself.
- Data storage specifications, especially the selection of the DBMS, are determined by business requirements. Make sure you collect enough relevant details during the requirements phase.
- Business requirements strongly influence the information delivery mechanism.
- Requirements define how, when, and where the users will receive information from the data warehouse.

Requirement gathering methods

Sources of Requirements

Good requirements start with good sources. Finding those quality sources is an important task and, fortunately, one that takes few resources. Examples of sources of requirements include:

- Customers
- Users
- Administrators and maintenance staff
- Partners
- Domain Experts
- Industry Analysts
- Information about competitors

Requirements Gathering Techniques

After you have identified these sources, there are a number of techniques that may be used to gather requirements. The following will describe the various techniques, followed by a brief discussion of when to use each technique.

To get the requirements down on paper, you can to do one or more of the following:

- Conduct a brainstorming session
- Interview users
- Send questionnaires
- Work in the target environment
- Study analogous systems
- Examine suggestions and problem reports
- Talk to support teams
- Study improvements made by users
- Look at unintended uses
- Conduct workshops
- Demonstrate prototypes to stakeholders

The best idea is to get the requirements down quickly and then to encourage the users to correct and improve them. Put in those corrections, and repeat the cycle. Do it now, keep it small, and correct it at once. Start off with the best structure you can devise, but expect

to keep on correcting it throughout the process. Success tips: Do it now, keep it small, and correct it immediately.

Requirement analysis and definition

Requirements analysis in systems engineering and software engineering, encompasses those tasks that go into determining the needs or conditions to meet for a new or altered product, taking account of the possibly conflicting requirements of the various stakeholders, such as beneficiaries or users. It is an early stage in the more general activity of requirements engineering which encompasses all activities concerned with eliciting, analyzing, documenting, validating and managing software or system requirements.

Requirements analysis is critical to the success of a systems or software project. The requirements should be documented, actionable, measurable, testable, traceable, related to identified business needs or opportunities, and defined to a level of detail sufficient for system design.

Conceptually, requirements analysis includes three types of activities:

- Eliciting requirements: the task of identifying the various types of requirements from various sources including project documentation, (e.g. the project charter or definition), business process documentation, and stakeholder interviews. This is sometimes also called requirements gathering.
- Analyzing requirements: determining whether the stated requirements are clear, complete, consistent and unambiguous, and resolving any apparent conflicts.
- Recording requirements: Requirements may be documented in various forms, usually including a summary list and may include natural-language documents, use cases, user stories, or process specifications.

Requirements analysis can be a long and arduous process during which many delicate psychological skills are involved. New systems change the environment and relationships between people, so it is important to identify all the stakeholders, take into account all their needs and ensure they understand the implications of the new systems. Analysts can employ several techniques to elicit the requirements from the customer. These may include the development of scenarios (represented as user stories in agile methods), the identification of use cases, the use of workplace observation or ethnography, holding interviews, or focus groups (more aptly named in this context as requirements workshops, or requirements review sessions) and creating requirements lists. Prototyping may be used to develop an example system that can be demonstrated to stakeholders. Where necessary, the analyst will employ a combination of these methods to establish the exact requirements of the stakeholders, so that a system that meets the business needs is produced.

Stakeholder identification

See Stakeholder analysis for a discussion of business uses. Stakeholders (SH) are people or organizations (legal entities such as companies, standards bodies) that have a valid interest in the system. They may be affected by it either directly or indirectly. A major new emphasis in the 1990s was a focus on the identification of *stakeholders*. It is increasingly recognized that stakeholders are not limited to the organization employing the analyst. Other stakeholders will include:

- anyone who operates the system (normal and maintenance operators)
- anyone who benefits from the system (functional, political, financial and social beneficiaries)
- anyone involved in purchasing or procuring the system. In a mass-market product organization, product management, marketing and

sometimes sales act as surrogate consumers (mass-market customers) to guide development of the product
- organizations which regulate aspects of the system (financial, safety, and other regulators)
- people or organizations opposed to the system (negative stakeholders; see also Misuse case)
- organizations responsible for systems which interface with the system under design
- those organizations who integrate horizontally with the organization for whom the analyst is designing the system

Stakeholder interviews

Stakeholder interviews are a common technique used in requirement analysis. Though they are generally idiosyncratic in nature and focused upon the perspectives and perceived needs of the stakeholder, often this perspective deficiency has the general advantage of obtaining a much richer understanding of the stakeholder's unique business processes, decision-relevant business rules, and perceived needs. Consequently this technique can serve as a means of obtaining the highly focused knowledge that is often not elicited in Joint Requirements Development sessions, where the stakeholder's attention is compelled to assume a more cross-functional context, and the desire to avoid controversy may limit the stakeholders willingness to contribute. Moreover, the in-person nature of the interviews provides a more relaxed environment where lines of thought may be explored at length.

Joint Requirements Development (JRD) Sessions

Requirements often have cross-functional implications that are unknown to individual stakeholders and often missed or incompletely defined during stakeholder interviews. These cross-functional implications can be elicited by conducting JRD sessions in a controlled environment, facilitated by a trained facilitator, wherein stakeholders participate in

discussions to elicit requirements, analyze their details and uncover cross-functional implications. A dedicated scribe and Business Analyst should be present to document the discussion. Utilizing the skills of a trained facilitator to guide the discussion frees the Business Analyst to focus on the requirements definition process.

JRD Sessions are analogous to Joint Application Design Sessions. In the former, the sessions elicit requirements that guide design, whereas the latter elicit the specific design features to be implemented in satisfaction of elicited requirements.

Contract-style requirement lists

One traditional way of documenting requirements has been contract style requirement lists. In a complex system such requirements lists can run to hundreds of pages long.

An appropriate metaphor would be an extremely long shopping list. Such lists are very much out of favour in modern analysis; as they have proved spectacularly unsuccessful at achieving their aims; but they are still seen to this day.

Strengths

- Provides a checklist of requirements.
- Provide a contract between the project sponsor(s) and developers.
- For a large system can provide a high level description.

Weaknesses

- Such lists can run to hundreds of pages. They are not intended to serve as a reader-friendly description of the desired application.
- Such requirements lists abstract all the requirements and so there is little context. The Business Analyst may include context for requirements in accompanying design documentation.
    - This abstraction is not intended to describe how the requirements fit or work together.

    - The list may not reflect relationships and dependencies between requirements. While a list does make it easy to prioritize each individual item, removing one item out of context can render an entire use case or business requirement useless.
    - The list doesn't supplant the need to review requirements carefully with stakeholders in order to gain a better shared understanding of the implications for the design of the desired system / application.
- Simply creating a list does not guarantee its completeness. The Business Analyst must make a good faith effort to discover and collect a substantially comprehensive list, and rely on stakeholders to point out missing requirements.
- These lists can create a false sense of mutual understanding between the stakeholders and developers; Business Analysts are critical to the translation process.
- It is almost impossible to uncover all the functional requirements before the process of development and testing begins. If these lists are treated as an immutable contract, then requirements that emerge in the Development process may generate a controversial change request.

Alternative to requirement lists

As an alternative to the requirement lists Agile Software Development uses User stories to suggest requirement in every day language.

Measurable goals

Best practices take the composed list of requirements merely as clues and repeatedly ask "why?" until the actual business purposes are discovered. Stakeholders and developers can then devise tests to measure what level of each goal has been achieved thus far. Such goals change more slowly than the long list of specific but unmeasured requirements. Once a small set of critical, measured goals has been established, rapid prototyping and short iterative development phases may proceed to deliver actual stakeholder value long before the project is half over.

Prototypes

Prototypes are Mockups of an application, allowing users to visualize an application that has not yet been constructed. Prototypes help people get an idea of what the system will look like, and make it easier for projects to make design decisions without waiting for the system to be built. Major improvements in communication between users and developers were often seen with the introduction of prototypes. Early views of applications led to fewer changes later and hence reduced overall costs considerably.

Prototypes can be flat diagrams (often referred to as wireframes) or working applications using synthesized functionality. Wireframes are made in a variety of graphic design documents, and often remove all color from the design (i.e. use a greyscale color palette) in instances where the final software is expected to have graphic design applied to it. This helps to prevent confusion as to whether the prototype represents the final visual look and feel of the application.

Use cases

A use case is a structure for documenting the functional requirements for a system, usually involving software, whether that is new or being changed. Each use case provides a set of *scenarios* that convey how the system should interact with a human user or another system, to achieve a specific business goal. Use cases typically avoid technical jargon, preferring instead the language of the end-user or *domain expert*. Use cases are often co-authored by requirements engineers and stakeholders.

Use cases are deceptively simple tools for describing the behavior of software or systems. A use case contains a textual description of the ways in which users are intended to work with the software or system. Use cases should not describe internal workings of the system, nor should they explain how that system will be implemented. Instead, they show the steps needed to perform a task.

Requirement Gathering and Analysis

The requirement gathering part is partially done in problem analysis phase. The document created in first step would help you to get started in this page. This is tough phase and is very critical to the success of your data warehouse development. Clear requirement and thorough analysis is must for success of any data warehouse project.

1. Prepare the questionnaire for the future users of the new Data Warehouse and Business Intelligence Solution.
2. Ask very specific questions rather than high level.
3. Find out the Data Sources for the Data Warehouse data.
3. Check if you can use existing system if any as a data source for the Data Warehouse.
4. Find out the possible data to be inserted every day in the Data Warehouse.
5. Get a list of reports that is supposed to come out of the new data warehouse.
6. Understand the existing Business and terminologies.
7. Find out if client is ready for commercial BO tools or Open Source tools.

Requirement analysis and gathering is a bit tough as client might no be clear with his/her requirement. It’s your expertise in the design process and prior experience which help you to analyze the requirement and put it on paper. Once everything is clear make sure you get a sign off from the client on the requirement.

Problem Analysis and Definition

Problem definition and analysis is the important phase of any development project. A good problem analysis and definition leads to a good solution.
While defining problems asking questions is a good practice to go to

deep into problems this help further while designing approaches to solve the problem.

Following questions might help you get started on this stage of Data warehouse design.

1. Is there any existing reporting system which is in use?
2. What are the main problem areas of existing system?
3. Interview the existing users to understand the problems and their expectation from future Business Intelligence Solution.

Document all your findings and now sit and analyze the problem. If possible try to put your problem in diagrammatic way and create a storyboard.

Design and Development

This is the actual start of your Data Warehouse project. Following things needs to executed in this step.

1. Design a data model based on requirement analysis. It could be star schema or snowflake schema based on requirement. Data modeling consist of following three parts.
a. Conceptual data model
b. Logical data model
c. Physical data model
2. Document in the data model design process including all possible details.
3. Get it reviewed and cross check the data model with reporting needs and your technical team.
5. Design a ETL process. ETL can be done using home grown tool or commercial tools. Having commercial tool speed up the process and also make is easy to maintain as skill set and support is available easily.
6. Create a sample data for the data warehouse and ETL. this could be a snap shot of the OLTP data.
5. Create a Unit testing plan and test ETL and data model for every new

addition and change.
6. Try creating sample reports on top of data warehouse to test it.

Prototyping

Many times this step can be skipped if Organization is ready to implement the data warehouse and clear about why they need a data warehouse. Following points can help you to successful execution of this step.

1. Get the most talented and experience people in this step as there past experience in data warehouse design will help a lot to design a prototype fast.
2. Make sure you implement the prototype in a manner which can be further reused in actual development.
3. Document the prototype
5. Use ready to use tools instead of using any high level language to create a tool to use in prototype. This will help to save time.
6. Create sample Data warehouse with sample data in it which can feed 2-3 reports.
7. Present the prototype to end users to get a feel of proposed solution.
8. Once done, Get sign off on prototype.

Remember prototype is a very small part of your actual development project. It needs to done fast. If Business Users like your prototype success of Data warehouse project is very near.

Principles of Dimensional Modelling

Dimensional Data Modeling

Dimensional Data Modeling is one of the data modeling techniques used in data warehouse design.

```
Goal: Improve the data retrieval.
```

The concept of Dimensional Modeling was developed by Ralph Kimball which is comprised of *facts and dimension* tables. Since the main goal of this modeling is to improve the data retrieval so it is optimized for *SELECT OPERATION*. The advantage of using this model is that we can store data in such a way that it is easier to store and retrieve the data once stored in a data warehouse. Dimensional model is the data model used by many OLAP systems.

Steps to Create Dimensional Data Modeling:

- **Step-1: Identifying the business objective –**
  The first step is to identify the business objective. Sales, HR, Marketing, etc. are some examples as per the need of the organization. Since it is the most important step of Data Modelling the selection of business objective also depends on the quality of data available for that process.
- **Step-2: Identifying Granularity –**
  Granularity is the lowest level of information stored in the table. The level of detail for business problem and its solution is described by Grain.
- **Step-3: Identifying Dimensions and its Attributes –**
  Dimensions are objects or things. Dimensions categorize and describe data warehouse facts and measures in a way that support meaningful answers to business questions. A data warehouse organizes descriptive attributes as columns in dimension tables. For Example, the data dimension may contain data like a year, month and weekday.

- **Step-4: Identifying the Fact –**
  The measurable data is hold by the fact table. Most of the fact table rows are numerical values like price or cost per unit, etc.
- **Step-5: Building of Schema –**
  We implement the Dimension Model in this step. A schema is a database structure. There are two popular schemes: Star Schema and Snowflake Schema.

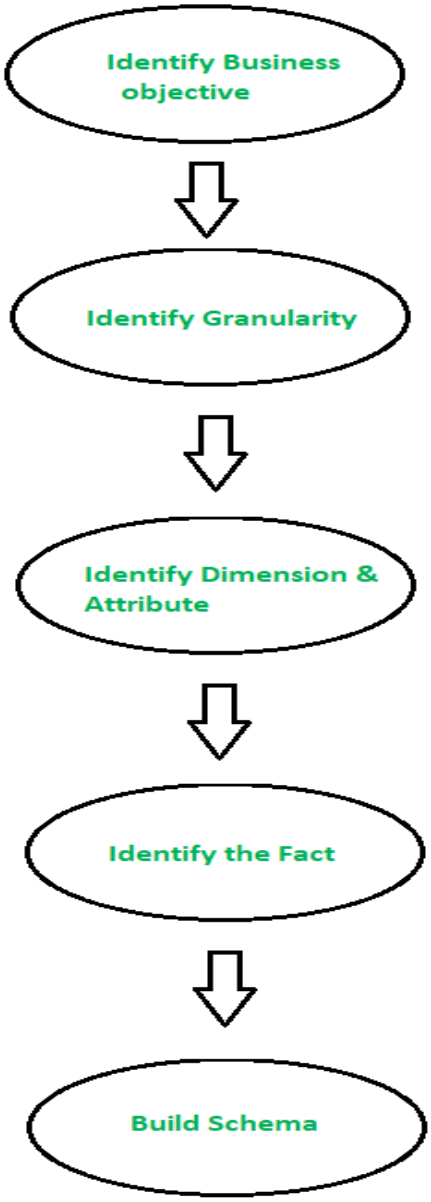


Figure – Steps for Dimensional Model

From requirements to data design

Good Business Intelligence (BI), allows your organization to query data obtained from trusted sources and use the answers to gain a competitive edge in your industry. The first step to achieving effective BI is a well-designed warehouse. Data warehouse design is the process of building a solution to integrate data from multiple sources that support analytical reporting and data analysis. A poorly designed data warehouse can result in acquiring and using inaccurate source data that negatively affect the productivity and growth of your organization. This blog post will take a high-level look at the data warehouse design process from requirements gathering to implementation.

Requirements Gathering

Gathering requirements is step one of the data warehouse design process. The goal of the requirements gathering phase is to determine the criteria for a successful implementation of the data warehouse. An organization's long-term business strategy should be just as important as the current business and technical requirements. User analysis and reporting requirements must be identified as well as hardware, development, testing, implementation, and user training.

Once the business and technical strategy has been decided the next step is to address how the organization will backup the data warehouse and how it will recover if the system fails. Developing a disaster recovery plan while gathering requirements, ensures that the organization is prepared to respond quickly to direct and indirect threats to the data warehouse.

Physical Environment Setup

Once the business requirements are set, the next step is to determine the physical environment for the data warehouse. At a minimum, there should be separate physical application and database servers as well as separate ETL/ELT, OLAP, cube, and reporting processes set up for development, testing, and production. Building separate physical environments ensure that all changes can be tested before moving them to production, development, and testing can occur without halting the production environment, and if data integrity becomes suspect, the IT staff can

investigate the issue without negatively impacting the production environment.

Data Modeling

Once requirements gathering and physical environments have been defined, the next step is to define how data structures will be accessed, connected, processed, and stored in the data warehouse. This process is known as data modeling. During this phase of data warehouse design, is where data sources are identified. Knowing where the original data resides and just as importantly, the availability of that data, is crucial to the success of the project. Once the data sources have been identified, the data warehouse team can begin building the logical and physical structures based on established requirements.

ETL

The ETL process takes the most time to develop and eats up the majority of implementation. Identifying data sources during the data modeling phase may help to reduce ETL development time. The goal of ETL is to provide optimized load speeds without sacrificing quality. Failure at this stage of the process can lead to poor performance of the ETL process and the entire data warehouse system.

OLAP Cube Design

On-Line Analytical Processing (OLAP) is the answer engine that provides the infrastructure for ad-hoc user query and multi-dimensional analysis. OLAP design specification should come from those who will query the data. Documentation specifying the OLAP cube dimensions and measures should be obtained during the beginning of data warehouse design process. The three critical elements of OLAP design include:

Grouping measures - numerical values you want to analyze such as revenue, number of customers, how many products customers purchase, or average purchase amount.

Dimension - where measures are stored for analysis such as geographic region, month, or quarter.

Granularity - the lowest level of detail that you want to include in the OLAP dataset.

During development, make sure the OLAP cube process is optimized. A data warehouse is usually not a nightly priority run, and once the data warehouse has been updated, there little time left to update the OLAP cube. Not updating either of them in a timely manner could lead to reduced system performance. Taking the time to explore the most efficient OLAP cube generation path can reduce or prevent performance problems after the data warehouse goes live.

Front End Development

At this point, business requirements have been captured, physical environment complete, data model decided, and ETL process has been documented. The next step is to work on how users will access the data warehouse. Front end development is how users will access the data for analysis and run reports. There are many options available, including building your front end in-house or purchasing an off the shelf product. Either way, there are a few considerations to keep in mind to ensure the best experience for end users.

Secure access to the data from any device - desktop, laptop, tablet, or phone should be the primary consideration. The tool should allow your development team to modify the backend structure as enterprise level reporting requirements change. It should also provide a Graphical User Interface (GUI) that enables users to customize their reports as needed. The OLAP engine and data can be the best in class, but if users are not able to use the data, the data warehouse becomes an expensive and useless data repository.

Report Development

For most end users, the only contact they have with the data warehouse is through the reports they generate. As mentioned in the front end development section, users' ability to select their report criteria quickly and efficiently is an essential feature for data warehouse report generation. Delivery options are another consideration. Along with receiving reports through a secure web interface, users may want or need reports sent as an email attachment, or spreadsheet. Controlling the flow and visibility of data is another aspect of report development that must be addressed. Developing user groups with access to specific data segments should provide data security and control. Reporting will and should change well

after the initial implementation. A well-designed data warehouse should be able to handle the new reporting requests with little to no data warehouse system modification.

Performance Tuning

Earlier in this post, the recommendation was to create separate development and testing environments. Doing so allows organizations to provide system performance tuning on ETL, query processing, and report delivery without interrupting the current production environment. Make sure the development and testing environments-hardware and applications mimic the production environment so that the performance enhancements created in development will work in the live production environment.

Testing

Once the data warehouse system has been developed according to business requirements, the next step is to test it. Testing, or quality assurance, is a step that should not be skipped because it will allow the data warehouse team to expose and address issues before the initial rollout. Failing to complete the testing phase could lead to implementation delays or termination of the data warehouse project.

Implementation

Time to go live. Deciding to make the system available to everyone at once or perform a staggered release, will depend on the number of end users and how they will access the data warehouse system. Another important aspect of any system implementation and one that is often skipped, is end-user training. No matter how "intuitive" the data warehouse team and developers think the GUI is, if the actual end users finds the tool difficult to use, or do not understand the benefits of using the data warehouse for reporting and analysis, they will not engage.

Understanding Best Practices for Data Warehouse Design

Data warehouse design is a time consuming and challenging endeavor. There will be good, bad, and ugly aspects found in each step. However, if an organization takes the time to develop sound requirements at the beginning, subsequent steps in the process will flow more logically and lead to a successful data warehouse implementation.

Multidimensional Data Model

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

A data cube allows data to be viewed in multiple dimensions. Dimensions 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.

What is Multidimensional schemas?

Multidimensional schema is especially designed to model data warehouse systems. The schemas are designed to address the unique needs of very large databases designed for the analytical purpose (OLAP).

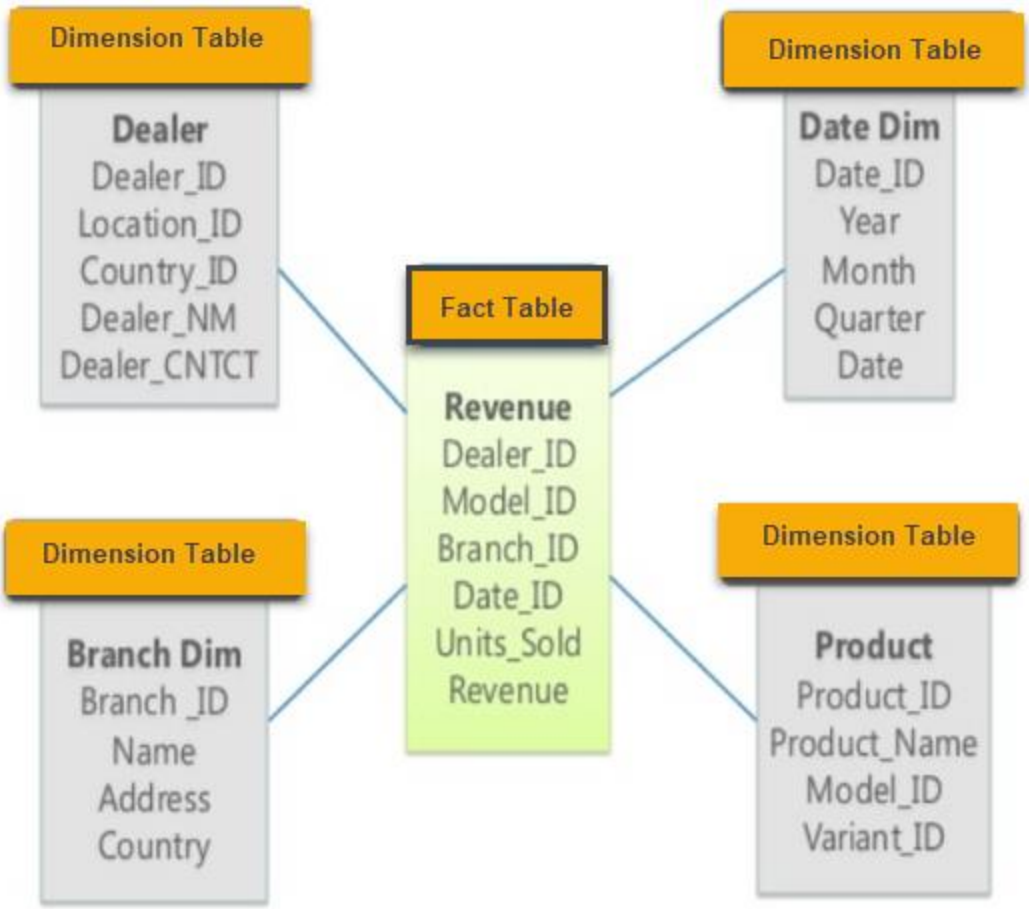
Types of Data Warehouse Schema:

Following are 3 chief types of multidimensional schemas each having its unique advantages.

- Star Schema
- Snowflake Schema
- Galaxy Schema

What is a Star Schema?

The star schema is the simplest type of Data Warehouse schema. It is known as star schema as its structure resembles a star. In the Star schema, the center of the star can have one fact tables and numbers of associated dimension tables. It is also known as Star Join Schema and is optimized for querying large data sets.

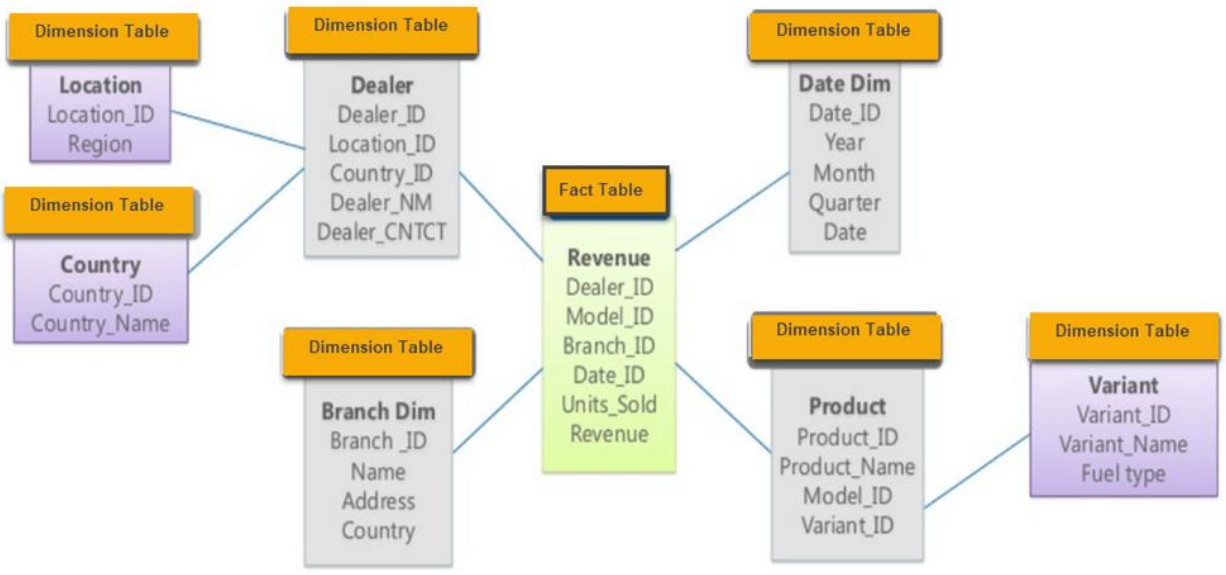


For example, as you can see in the above-given image that fact table is at the center which contains keys to every dimension table like Deal_ID, Model ID, Date_ID, Product_ID, Branch_ID & other attributes like Units sold and revenue.

Characteristics of Star Schema:

- Every dimension in a star schema is represented with the only one-dimension table.
- The dimension table should contain the set of attributes.
- The dimension table is joined to the fact table using a foreign key
- The dimension table are not joined to each other
- Fact table would contain key and measure
- The Star schema is easy to understand and provides optimal disk usage.
- The dimension tables are **not normalized**. For instance, in the above figure, Country_ID does not have Country lookup table as an OLTP design would have.
- The schema is widely supported by BI Tools

What is a Snowflake Schema?



A Snowflake Schema is an extension of a Star Schema, and it adds additional dimensions. It is called snowflake because its diagram resembles a Snowflake.

The dimension tables are **normalized** which splits data into additional tables. In the following example, Country is further normalized into an individual table.

Characteristics of Snowflake Schema:

- The main benefit of the snowflake schema it uses smaller disk space.
- Easier to implement a dimension is added to the Schema
- Due to multiple tables query performance is reduced
- The primary challenge that you will face while using the snowflake Schema is that you need to perform more maintenance efforts because of the more lookup tables.

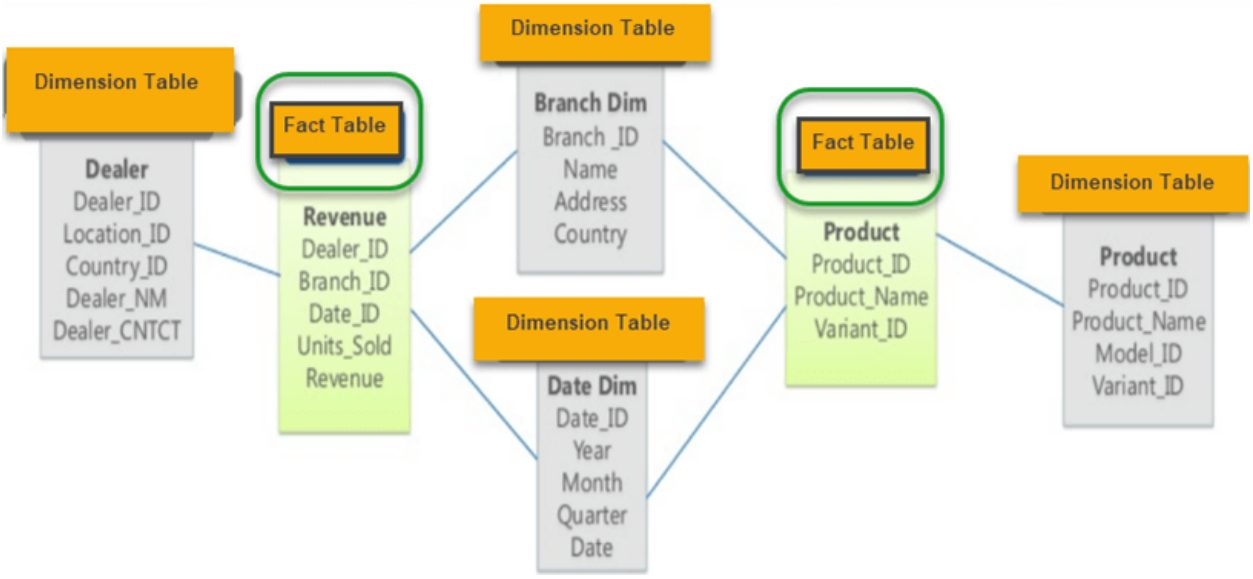
Star Vs Snowflake Schema: Key Differences

Star Schema	Snow Flake 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 Snow Flake Schema is represented by centralized fact table which unlikely connected with multiple dimensions.

What is a Galaxy schema(Fact Constellation Schema)?

A Galaxy Schema contains two fact table that shares dimension tables. It is also called Fact Constellation Schema. The schema is viewed as a collection of stars hence the name Galaxy Schema.



As you can see in above figure, there are two facts table

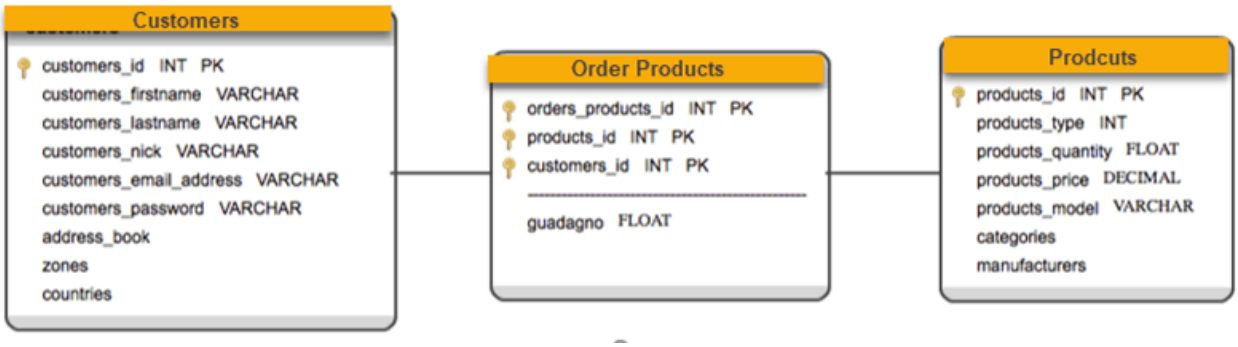
1. Revenue
2. Product.

In Galaxy schema shares dimensions are called Conformed Dimensions.

Characteristics of Galaxy Schema:

- The dimensions in this schema are separated into separate dimensions based on the various levels of hierarchy.
- For example, if geography has four levels of hierarchy like region, country, state, and city then Galaxy schema should have four dimensions.
- Moreover, it is possible to build this type of schema by splitting the one-star schema into more Star schemes.
- The dimensions are large in this schema which is needed to build based on the levels of hierarchy.
- This schema is helpful for aggregating fact tables for better understanding.

What is Star Cluster Schema?



Snowflake schema contains fully expanded hierarchies. However, this can add complexity to the Schema and requires extra joins. On the other hand, star schema contains fully collapsed hierarchies, which may lead to redundancy. So, the best solution may be a balance between these two schemas which is star cluster schema design.

Overlapping dimensions can be found as forks in hierarchies. A fork happens when an entity acts as a parent in two different dimensional hierarchies. Fork entities then identified as classification with one-to-many relationships.

OLAP in the Data Warehouse

Limitations of other analysis methods (OLTP)

Online transaction processing refers to more than just financial transactions. Telemarketers use OLTP for their phone surveys, call centers use it to access customer data and companies use it to manage customers' online accounts. Businesses also use it to facilitate e-commerce, their internal communications and numerous applications. If your company doesn't use don't use OLTP, it may lose sales opportunities and operational efficiencies. However, OLTP also comes with important disadvantages and limitations.

Unscheduled Downtime

Your business can suffer considerable losses when the OLTP system goes down, even temporarily. This can happen due to network outages, data corruption or hardware failure. Companies can protect their operation by building redundancy into the business platform, but that may not prove cost-effective for smaller businesses. To mitigate these concerns, hire competent IT personnel who are available 24/7 to respond to critical issues. You also must maintain good lines of communication with your institutional partners, so that you can get support when you need it and pass information along to others who may be affected.

Concurrency Challenges

OLTP systems allow multiple users to access and modify the same data at the same time. For obvious reasons, you can't allow one user to change data while another person is modifying it. You must

devise an efficient way to ensure people aren't working at cross purposes while retaining a system that is responsive for everyone. This may require costly systems designs and maintenance. OLTP concurrency best practices have evolved in step with the growth of the Internet and OLTP itself, so the solutions are available in the form of OLTP software packages, but if you can't implement them yourself, you must hire a professional to do it.

Atomicity

In OLTP, "atomicity" refers to a transaction in which either all the database steps succeed or the entire transaction fails. If any one step goes wrong, and the transaction continues anyway, you'll probably end up with data errors or corruption. That could be devastating for your company. All OLTP transactions should be atomic, with an emphasis on data recovery when something goes wrong. However, there may be bottom-line consequences when the technology doesn't work correctly. Inefficiently implemented database atomicity also may cause system slowdowns.

Financial Transaction Processing Costs

For many businesses, "OLTP" narrowly refers to transactions with financial institutions -- mainly credit and debit card payments over the Internet or through physical card readers. Financial institutions do impose costs on merchants for these transactions. Your business will be charged monthly fees, minimum fees, and gateway payments. Transaction fees hit you twice, first as a percentage of the value of the entire transaction, and then as a supplemental absolute fee that's usually a fraction of a dollar. According to figures collected by

Community Merchants USA and reported in Forbes, plastic transactions -- debit, credit and gift cards -- accounted for two-thirds of all point-of-sale transactions in 2013. Few businesses can afford to ignore these payment methods. You either must eat the costs, include the cost in your margins or charge customers a fee for paying electronically.

OLAP is the answer…

An effective OLAP solution solves problems for both business users and IT departments. For business users, it enables fast and intuitive access to centralized data and related calculations for the purposes of analysis and reporting. For IT, an OLAP solution enhances a data warehouse or other relational database with aggregate data and business calculations. In addition, by enabling business users to do their own analyses and reporting, OLAP systems reduce demands on IT resources.

OLAP offers five key benefits:

- Business-focused multidimensional data
- Business-focused calculations
- Trustworthy data and calculations
- Speed-of-thought analysis
- Flexible, self-service reporting

OLAP Definition and Rules

OLAP stands for "Online Analytical Processing." OLAP allows users to analyse database information from multiple database systems at one time. While relational databases are considered to be two-dimensional, OLAP data is multidimensional,

meaning the information can be compared in many different ways. For example, a company might compare their computer sales in June with sales in July, then compare those results with the sales from another location, which might be stored in a different database.

In order to process database information using OLAP, an OLAP server is required to organize and compare the information. Clients can analyse different sets of data using functions built into the OLAP server. Some popular OLAP server software programs include Oracle Express Server and Hyperion Solutions Essbase. Because of its powerful data analysis capabilities, OLAP processing is often used for data mining, which aims to discover new relationships between different sets of data.

Codd's rules for OLAP Tools

In 1993, Dr. E.F. Codd originated twelve rules as the basis for selecting OLAP tools. The publication of these rules was the result of research carried out on behalf of Arbor Software and has resulted in a formalized redefinition of the requirements for OLAP tools. These rules are:

1. Multi-dimensional conceptual view of the database
2. Concept of transparency
3. Concept of accessibility
4. Consistent reporting performance
5. Client-server architecture

6. Generic dimensionality
7. Dynamic sparse matrix handling
8. Multi-user support
9. Unrestricted cross-dimensional operations
10. Intuitive data manipulation
11. Flexible reporting
12. Unlimited dimensions and aggregation levels

- **Multi-dimensional conceptual view:** OLAP tools should allow users with a multi-dimensional model that keep up a correspondence to users' views of the enterprise and is intuitively analytical and simple to use. Interestingly, this rule is given various levels of support by sellers who disagree that a multi-dimensional conceptual view of data can be delivered without multi-dimensional storage.

- **Transparency:** The OLAP technology has the underlying database and architecture, and the likely heterogeneity of input data sources that should be apparent to users. This necessity is to preserve the user's productivity and proficiency with familiar front-end environments and tools.

- **Accessibility:** The OLAP tool also let to access data needed for the analysis from all heterogeneous enterprise data sources such as relational, non-relational, and legacy methods.

- **Consistent reporting performance:** With the number of dimensions, levels of aggregations, and the size of the database raises, users ought to not perceive any significant fall in performance. There should be no change in the way the key figures are calculated, and the system models must have to be strong enough to cope with changes to the enterprise model.

- **Client-server architecture:** The OLAP system should be proficient enough to operate efficiently in a client-server environment. The architecture should permit optimal performance, flexibility, adaptability, scalability, and interoperability.
- **Generic dimensionality:** Every data dimension must be the same in both structure and operational capabilities, i.e., the basic structure, formulae, and reporting should not be biased towards any one dimension.
- **Dynamic sparse matrix handling:** The OLAP system should be able to cope up with the physical schema to the specific analytical model that optimizes sparse matrix handling to achieve and maintain the required level of performance.
- **Multi-user support:** The OLAP system should be able to hold up a group of users working at the same time on the same or different models of the enterprise's data.
- **Unrestricted cross-dimensional operations:** The OLAP system must be able to identify the dimensional hierarchies and automatically perform associated roll-up calculations across dimensions.
- **Intuitive data manipulation:** Slicing and cubing, consolidation (roll-up), and other manipulations can be accomplished via direct 'point-and-click' or 'drag-and-drop' actions on the cells of the cube.
- **Flexible reporting:** The capability of arranging rows, columns, and cells in a Way that facilitates analysis by an intuitive visual presentation of analytical reports must exist.

- **Unlimited dimensions and aggregation levels:** Depending on business needs, an analytical model may have some dimensions each having multiple hierarchies.

Characteristics of OLAP

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.

Features of OLAP

Online Analytical Processing (OLAP) is usually contrasted from Online Transactional Processing (OLTP) and is a way of storing data so that it can be used for better analytical queries. The biggest difference in OLAP is that data is stored in what is commonly referred to as an OLAP cube. In OLTP however, the data is stored in tables and it uses a relational database system.

What OLAP does is pre-process all of the ways someone might want to analyze the data so that when someone does use it, the process is very fast.

OLAP provides summary data and generates rich calculations. For example, OLAP answers questions like "How do sales of

mutual funds in North America for this quarter compare with sales a year ago? What can we predict for sales next quarter? What is the trend as measured by percent change?"

Advantages of OLAP

- Fast query performance due to optimized storage, multidimensional indexing and caching.
- Smaller on-disk size of data compared to data stored in relational database due to compression techniques.
- Automated computation of higher level aggregates of the data.
- It is very compact for low dimension data sets.
- Array models provide natural indexing.
- Effective data extraction achieved through the pre-structuring of aggregated data.

Disadvantages of OLAP

- Within some OLAP Solutions the processing step (data load) can be quite lengthy, especially on large data volumes. This is usually remedied by doing only incremental processing, i.e., processing only the data which have changed (usually new data) instead of reprocessing the entire data set.
- Some OLAP methodologies introduce data redundancy.

Functions of OLAP

OLAP can be used for data mining or the discovery of previously undiscerned relationships between data items. An OLAP database does not need to be as large as a data warehouse,

since not all transactional data is needed for trend analysis. Using Open Database Connectivity (ODBC), data can be imported from existing relational databases to create a multidimensional database for OLAP.

OLAP products include IBM Cognos, Oracle OLAP and Oracle Essbase. OLAP features are also included in tools such as Microsoft Excel and Microsoft SQL Server's Analysis Services). OLAP products are typically designed for multiple-user environments, with the cost of the software based on the number of users.

Hypercube (OLAP Cube)

An OLAP cube is a multidimensional database that is optimized for data warehouse and online analytical processing (OLAP) applications.

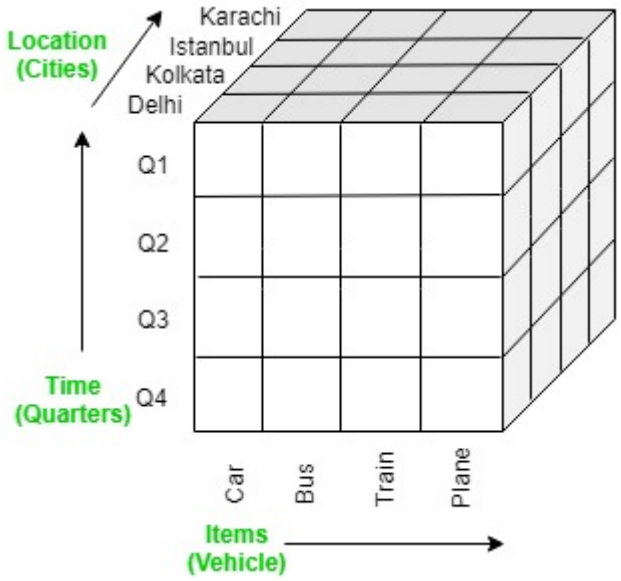
An OLAP cube is a method of storing data in a multidimensional form, generally for reporting purposes. In OLAP cubes, data (measures) are categorized by dimensions. OLAP cubes are often pre-summarized across dimensions to drastically improve query time over relational databases. The query language used to interact and perform tasks with OLAP cubes is multidimensional expressions (MDX). The MDX language was originally developed by Microsoft in the late 1990s, and has been adopted by many other vendors of multidimensional databases.

Although it stores data like a traditional database does, an OLAP cube is structured very differently. Databases, historically, are designed according to the requirements of the IT systems that use them. OLAP cubes, however, are used by business users for advanced analytics. Thus, OLAP cubes are designed using business logic and understanding. They are optimized for

analytical purposes, so that they can report on millions of records at a time. Business users can query OLAP cubes using plain English.

OLAP Operations

OLAP stands for ***Online Analytical Processing*** Server. It is a software technology that allows users to analyze information from multiple database systems at the same time. It is based on multidimensional data model and allows the user to query on multi-dimensional data (eg. Delhi -> 2018 -> Sales data). OLAP databases are divided into one or more cubes and these cubes are known as *Hyper-cubes*.

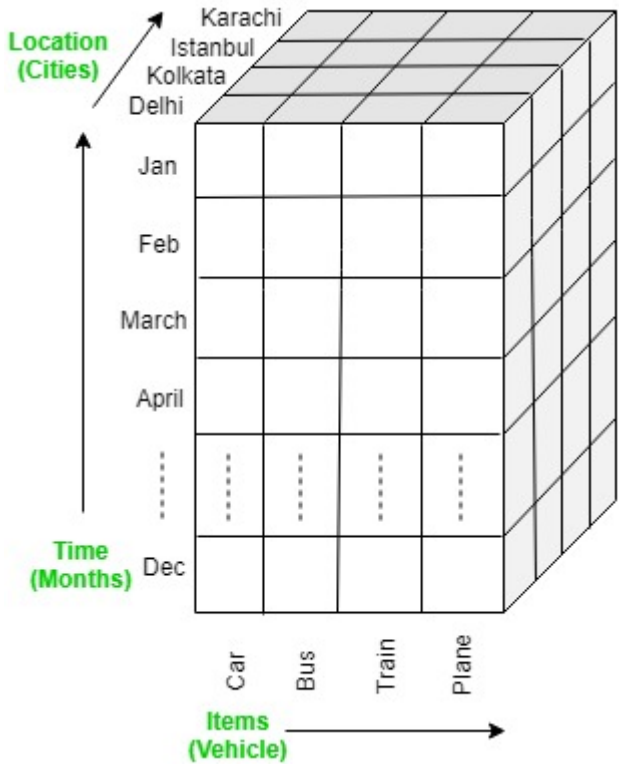


OLAP operations:

There are five basic analytical operations that can be performed on an OLAP cube:

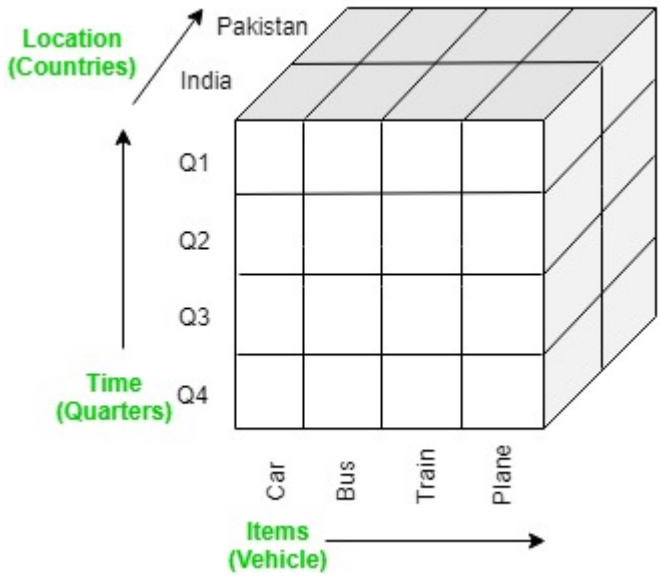
1. **Drill down:** In drill-down operation, the less detailed data is converted into highly detailed data. It can be done by:
   - Moving down in the concept hierarchy
   - Adding a new dimension

In the cube given in overview section, the drill down operation is performed by moving down in the concept hierarchy of *Time* dimension (Quarter -> Month).

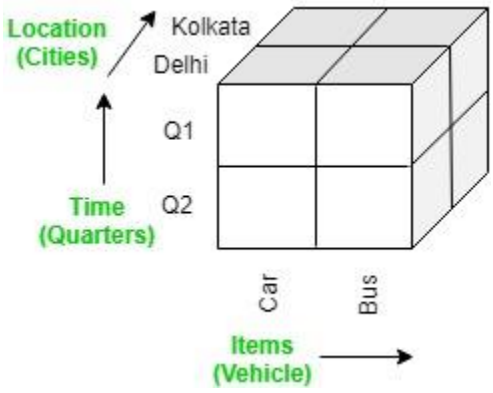


2. **Roll up:** It is just opposite of the drill-down operation. It performs aggregation on the OLAP cube. It can be done by:
    - Climbing up in the concept hierarchy
    - Reducing the dimensions

In the cube given in the overview section, the roll-up operation is performed by climbing up in the concept hierarchy of Location dimension (City -> Country).

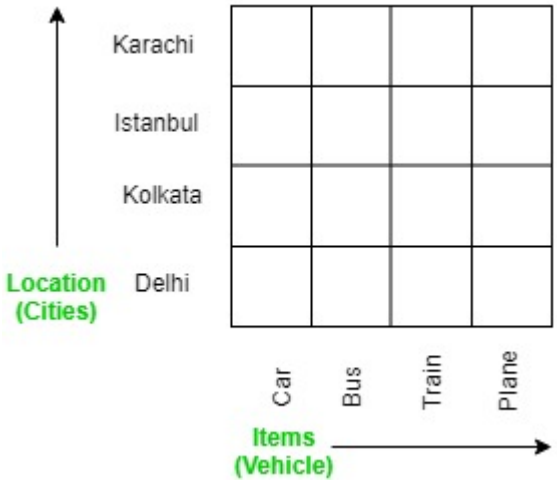


3. **Dice:** It selects a sub-cube from the OLAP cube by selecting two or more dimensions. In the cube given in the overview section, a sub-cube is selected by selecting following dimensions with criteria:
   - Location = “Delhi” or “Kolkata”
   - Time = “Q1” or “Q2”
   - Item = “Car” or “Bus”

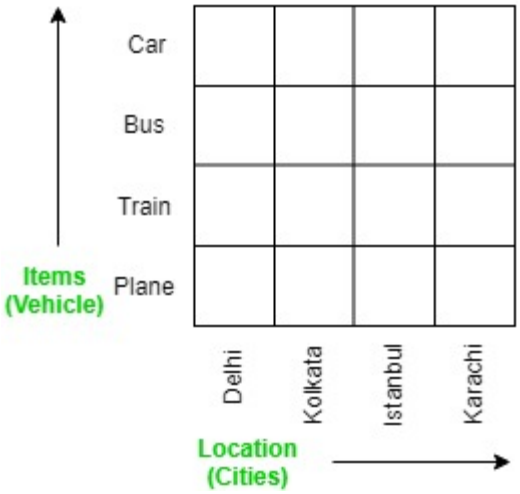


4. **Slice:** It selects a single dimension from the OLAP cube which results in a new sub-cube creation. In the cube given in the overview section, Slice is performed on the dimension Time =

“Q1”.

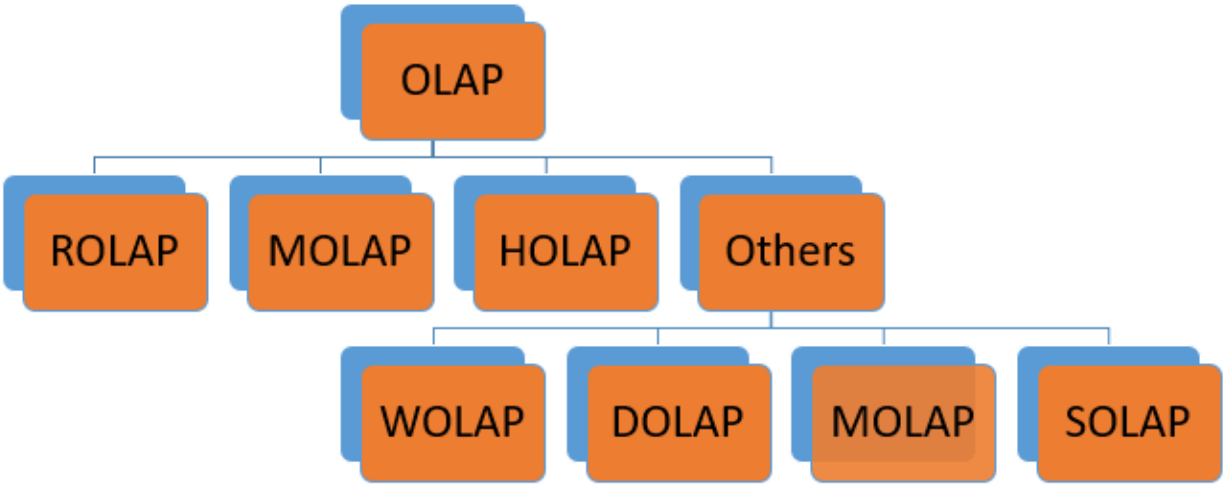


5. **Pivot:** It is also known as *rotation* operation as it rotates the current view to get a new view of the representation. In the sub-cube obtained after the slice operation, performing pivot operation gives a new view of it.



Types of OLAP systems

OLAP Hierarchical Structure



Type of OLAP	Explanation
Relational OLAP(ROLAP):	ROLAP is an extended RDBMS along with multidimensional data mapping to perform the standard relational operation.
Multidimensional OLAP (MOLAP)	MOLAP Implementes operation in multidimensional data.
Hybrid OnlineAnalytical Processing (HOLAP)	In HOLAP approach the aggregated totals are stored in a multidimensional database while the detailed data is stored in the relational database. This offers both data efficiency of the ROLAP model and the performance of the MOLAP model.

Desktop OLAP (DOLAP)	In Desktop OLAP, a user downloads a part of the data from the database locally, or on their desktop and analyze it. DOLAP is relatively cheaper to deploy as it offers very few functionalities compares to other OLAP systems.
Web OLAP (WOLAP)	Web OLAP which is OLAP system accessible via the web browser. WOLAP is a three-tiered architecture. It consists of three components: client, middleware, and a database server.
Mobile OLAP:	Mobile OLAP helps users to access and analyze OLAP data using their mobile devices
Spatial OLAP :	SOLAP is created to facilitate management of both spatial and non-spatial data in a Geographic Information system (GIS)

ROLAP

ROLAP works with data that exist in a relational database. Facts and dimension tables are stored as relational tables. It also allows multidimensional analysis of data and is the fastest growing OLAP.

Advantages of ROLAP model:

- **High data efficiency.** It offers high data efficiency because query performance and access language are optimized particularly for the multidimensional data analysis.

- **Scalability.** This type of OLAP system offers scalability for managing large volumes of data, and even when the data is steadily increasing.

Drawbacks of ROLAP model:

- **Demand for higher resources:** ROLAP needs high utilization of manpower, software, and hardware resources.
- **Aggregately data limitations.** ROLAP tools use SQL for all calculation of aggregate data. However, there are no set limits to the for handling computations.
- **Slow query performance.** Query performance in this model is slow when compared with MOLAP

MOLAP

MOLAP uses array-based multidimensional storage engines to display multidimensional views of data. Basically, they use an OLAP cube.

Hybrid OLAP

Hybrid OLAP is a mixture of both ROLAP and MOLAP. It offers fast computation of MOLAP and higher scalability of ROLAP. HOLAP uses two databases.

1. Aggregated or computed data is stored in a multidimensional OLAP cube
2. Detailed information is stored in a relational database.

Benefits of Hybrid OLAP:

- This kind of OLAP helps to economize the disk space, and it also remains compact which helps to avoid issues related to access speed and convenience.
- Hybrid HOLAP's uses cube technology which allows faster performance for all types of data.
- ROLAP are instantly updated and HOLAP users have access to this real-time instantly updated data. MOLAP brings cleaning and conversion of data thereby improving data relevance. This brings best of both worlds.

Drawbacks of Hybrid OLAP:

- Greater complexity level**:** The major drawback in HOLAP systems is that it supports both ROLAP and MOLAP tools and applications. Thus, it is very complicated.
- Potential overlaps**:** There are higher chances of overlapping especially into their functionalities.

ROLAP vs MOLAP

BASIS FOR COMPARISON	ROLAP	MOLAP
Full Form	ROLAP stands for Relational Online Analytical Processing.	MOLAP stands for Multidimensional Online Analytical Processing.
Storage & Fetched	Data is stored and fetched from the main data warehouse.	Data is Stored and fetched from the Proprietary database MDDBs.
Data Form	Data is stored in the form of relational tables.	Data is Stored in the large multidimensional array made of data cubes.
Data volumes	Large data volumes.	Limited summaries data is kept in MDDBs.

BASIS FOR COMPARISON	ROLAP	MOLAP
Technology	Uses Complex SQL queries to fetch data from the main warehouse.	MOLAP engine created a precalculated and prefabricated data cubes for multidimensional data views. Sparse matrix technology is used to manage data sparsity.
View	ROLAP creates a multidimensional view of data dynamically.	MOLAP already stores the static multidimensional view of data in MDDBs.
Access	Slow access.	Faster access.

Advantages of OLAP

- OLAP is a platform for all type of business includes planning, budgeting, reporting, and analysis.
- Information and calculations are consistent in an OLAP cube. This is a crucial benefit.
- Quickly create and analyze "What if" scenarios
- Easily search OLAP database for broad or specific terms.

- OLAP provides the building blocks for business modeling tools, Data mining tools, performance reporting tools.
- Allows users to do slice and dice cube data all by various dimensions, measures, and filters.
- It is good for analyzing time series.
- Finding some clusters and outliers is easy with OLAP.
- It is a powerful visualization online analytical process system which provides faster response times

Disadvantages of OLAP

- OLAP requires organizing data into a star or snowflake schema. These schemas are complicated to implement and administer
- You cannot have large number of dimensions in a single OLAP cube
- Transactional data cannot be accessed with OLAP system.
- Any modification in an OLAP cube needs a full update of the cube. This is a time-consuming process

Executive Information System (EIS)

Definition - What does Executive Information System (EIS) mean?

An executive information system (EIS) is a decision support system (DSS) used to assist senior executives in the decision-making process. It does this by providing easy access to important data needed to achieve strategic goals in an organization. An EIS normally features graphical displays on an easy-to-use interface.

Executive information systems can be used in many different types of organizations to monitor enterprise performance as well as to identify opportunities and problems.

Data Warehouse and Business Strategy

This service can be targeted at producing a new Data Warehouse (DW) strategy or for working within your company's existing DW strategy. Both business and IT management will be interviewed by the Data Warehouse Architect to develop a well-grounded strategy for the long-term and short-term.
We assess your data warehouse and business intelligence needs. Then we develop a prioritized, high level plan on how to reach your goals.

Business Analysis

Working from the results of the Opportunity Assessment, and using the Data Warehouse Strategy as a framework, the Senior Business Analyst identifies the detailed requirements for the solution.
The Senior Business Analyst works with end users to capture their intended use of the information for business growth, determine what information they need to see, and how they need to see it:

- Identify the Key Performance Indicators (KPIs) which will be used to track objectives.
- Online analysis processing requirements (analytics).
- Security requirements.
- Review current reporting systems that report on related data.
- Describe reports that need to be provided.
- Span of history to be kept in the data mart.
- Facts and Dimensions required to support the analysis and reports.
- Frequency of generating analytics reports.
- Frequency of refreshing the data mart.
- Geographic location of users.
- Method of distributing reports.
- Estimate how much data will be processed and stored for facts, dimensions, summaries and indexes.

Outputs from this service include:

- Documentation of objectives, KPIs, requirements, reports, estimates, assumptions and findings

- High level Logical Data Model.
- Scenarios describing how the analytics and reports will be used in the business context.

Based on findings from the Business Analysis service, the Data Warehouse Architect works with the company's Data Administrator to determine the data sources that are required to support the business needs, and the quality of the data in those sources.

Sources may be internal or external. Internal data sources typically belong to the company and reside in operational databases such as an order processing database. External data sources may be acquired from outside the organization, and often are used for demographic analyses or database marketing.

If similar data exists in more than one source, we work with the Data Administrator and business user to determine the single authoritative source

Data quality issues may include:

- Redundant data sources
- Duplicate rows within a data source
- Dirty data (invalid values)
- Data Integrity problems (inconsistent meanings and usage of fields)
- Referential Integrity problems (missing links, invalid links)

We will recommend solutions for addressing data quality issues. It is imperative that the best quality data be used to drive your analytics.

Data quality improvement may involve one-time and ongoing activities such as removing duplicates, standardizing names and addresses, deriving implied values, and improving data validation or logic in the operational system which generates the data.

UNIT 3

Data Mining Basics

What is Data Mining?

Data mining is looking for hidden, valid, and potentially useful patterns in huge data sets. Data Mining is all about discovering unsuspected/ previously unknown relationships amongst the data.

It is a multi-disciplinary skill that uses machine learning, statistics, AI and database technology.

The insights derived via Data Mining can be used for marketing, fraud detection, and scientific discovery, etc.

Data mining is also called as Knowledge discovery, Knowledge extraction, data/pattern analysis, information harvesting, etc.

Types of Data

Data mining can be performed on following types of data

- Relational databases
- Data warehouses
- Advanced DB and information repositories
- Object-oriented and object-relational databases
- Transactional and Spatial databases
- Heterogeneous and legacy databases
- Multimedia and streaming database
- Text databases
- Text mining and Web mining

Data Mining Implementation Process

Let's study the Data Mining implementation process in detail

Business understanding:

In this phase, business and data-mining goals are established.

- First, you need to understand business and client objectives. You need to define what your client wants (which many times even they do not know themselves)
- Take stock of the current data mining scenario. Factor in resources, assumption, constraints, and other significant factors into your assessment.
- Using business objectives and current scenario, define your data mining goals.
- A good data mining plan is very detailed and should be developed to accomplish both business and data mining goals.

Data understanding:

In this phase, sanity check on data is performed to check whether its appropriate for the data mining goals.

- First, data is collected from multiple data sources available in the organization.
- These data sources may include multiple databases, flat filer or data cubes. There are issues like object matching and schema integration which can arise during Data Integration process. It is a quite complex and tricky process as data from various sources unlikely to match easily. For example, table A contains an entity named cust_no whereas another table B contains an entity named cust-id.
- Therefore, it is quite difficult to ensure that both of these given objects refer to the same value or not. Here, Metadata should be used to reduce errors in the data integration process.
- Next, the step is to search for properties of acquired data. A good way to explore the data is to answer the data mining questions (decided in business phase) using the query, reporting, and visualization tools.
- Based on the results of query, the data quality should be ascertained. Missing data if any should be acquired.

Data preparation:

In this phase, data is made production ready.

The data preparation process consumes about 90% of the time of the project.

The data from different sources should be selected, cleaned, transformed, formatted, anonymized, and constructed (if required).

Data cleaning is a process to "clean" the data by smoothing noisy data and filling in missing values.

For example, for a customer demographics profile, age data is missing. The data is incomplete and should be filled. In some cases, there could be data outliers. For instance, age has a value 300. Data could be inconsistent. For instance, name of the customer is different in different tables.

Data transformation operations change the data to make it useful in data mining. Following transformation can be applied

Data transformation:

Data transformation operations would contribute toward the success of the mining process.

Smoothing: It helps to remove noise from the data.

Aggregation: Summary or aggregation operations are applied to the data. I.e., the weekly sales data is aggregated to calculate the monthly and yearly total.

Generalization: In this step, Low-level data is replaced by higher-level concepts with the help of concept hierarchies. For example, the city is replaced by the county.

Normalization: Normalization performed when the attribute data are scaled up o scaled down. Example: Data should fall in the range -2.0 to 2.0 post-normalization.

Attribute construction: these attributes are constructed and included the given set of attributes helpful for data mining.

The result of this process is a final data set that can be used in modeling.

Modelling

In this phase, mathematical models are used to determine data patterns.

- Based on the business objectives, suitable modeling techniques should be selected for the prepared dataset.
- Create a scenario to test check the quality and validity of the model.
- Run the model on the prepared dataset.
- Results should be assessed by all stakeholders to make sure that model can meet data mining objectives.

Evaluation:

In this phase, patterns identified are evaluated against the business objectives.

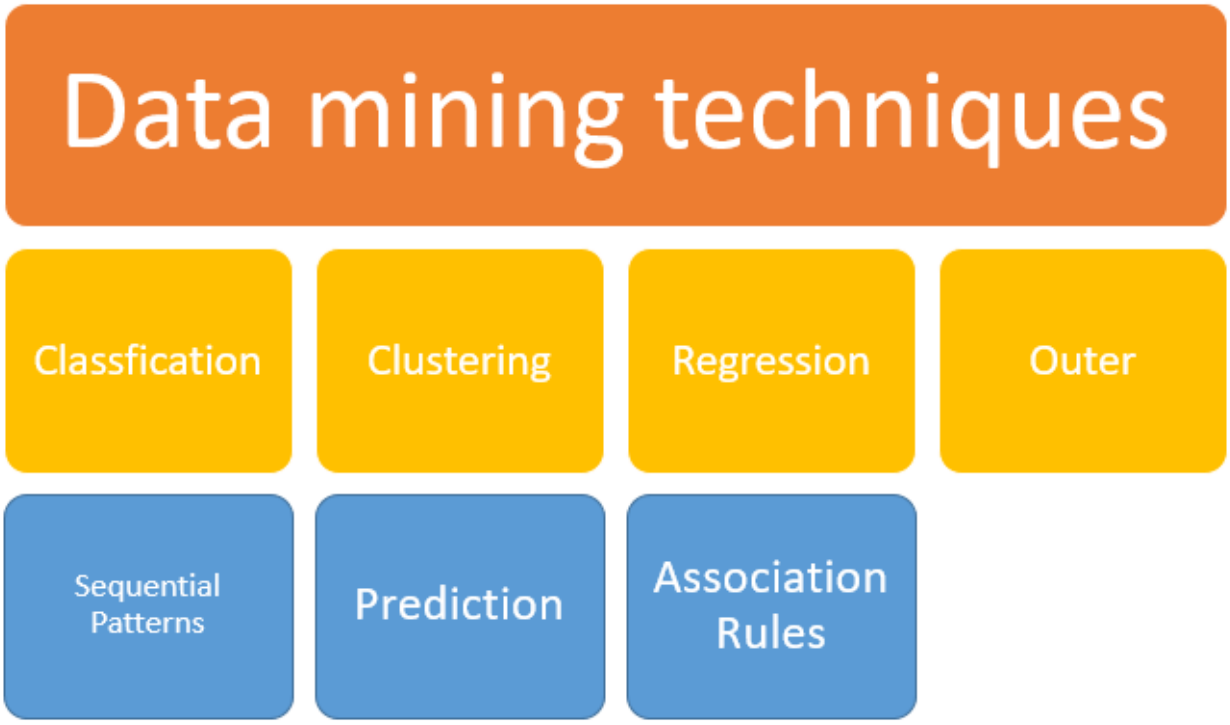
- Results generated by the data mining model should be evaluated against the business objectives.
- Gaining business understanding is an iterative process. In fact, while understanding, new business requirements may be raised because of data mining.
- A go or no-go decision is taken to move the model in the deployment phase.

Deployment:

In the deployment phase, you ship your data mining discoveries to everyday business operations.

- The knowledge or information discovered during data mining process should be made easy to understand for non-technical stakeholders.
- A detailed deployment plan, for shipping, maintenance, and monitoring of data mining discoveries is created.
- A final project report is created with lessons learned and key experiences during the project. This helps to improve the organization's business policy.

Data Mining Techniques



1.Classification:

This analysis is used to retrieve important and relevant information about data, and metadata. This data mining method helps to classify data in different classes.

2. Clustering:

Clustering analysis is a data mining technique to identify data that are like each other. This process helps to understand the differences and similarities between the data.

3. Regression:

Regression analysis is the data mining method of identifying and analyzing the relationship between variables. It is used to identify the likelihood of a specific variable, given the presence of other variables.

4. Association Rules:

This data mining technique helps to find the association between two or more Items. It discovers a hidden pattern in the data set.

5. Outer detection:

This type of data mining technique refers to observation of data items in the dataset which do not match an expected pattern or expected behavior. This technique can be used in a variety of domains, such as intrusion, detection, fraud or fault detection, etc. Outer detection is also called Outlier Analysis or Outlier mining.

6. Sequential Patterns:

This data mining technique helps to discover or identify similar patterns or trends in transaction data for certain period.

7. Prediction:

Prediction has used a combination of the other data mining techniques like trends, sequential patterns, clustering, classification, etc. It analyzes past events or instances in a right sequence for predicting a future event.

Challenges of Implementation of Data mine:

- Skilled Experts are needed to formulate the data mining queries.
- Overfitting: Due to small size training database, a model may not fit future states.
- Data mining needs large databases which sometimes are difficult to manage
- Business practices may need to be modified to determine to use the information uncovered.
- If the data set is not diverse, data mining results may not be accurate.
- Integration information needed from heterogeneous databases and global information systems could be complex

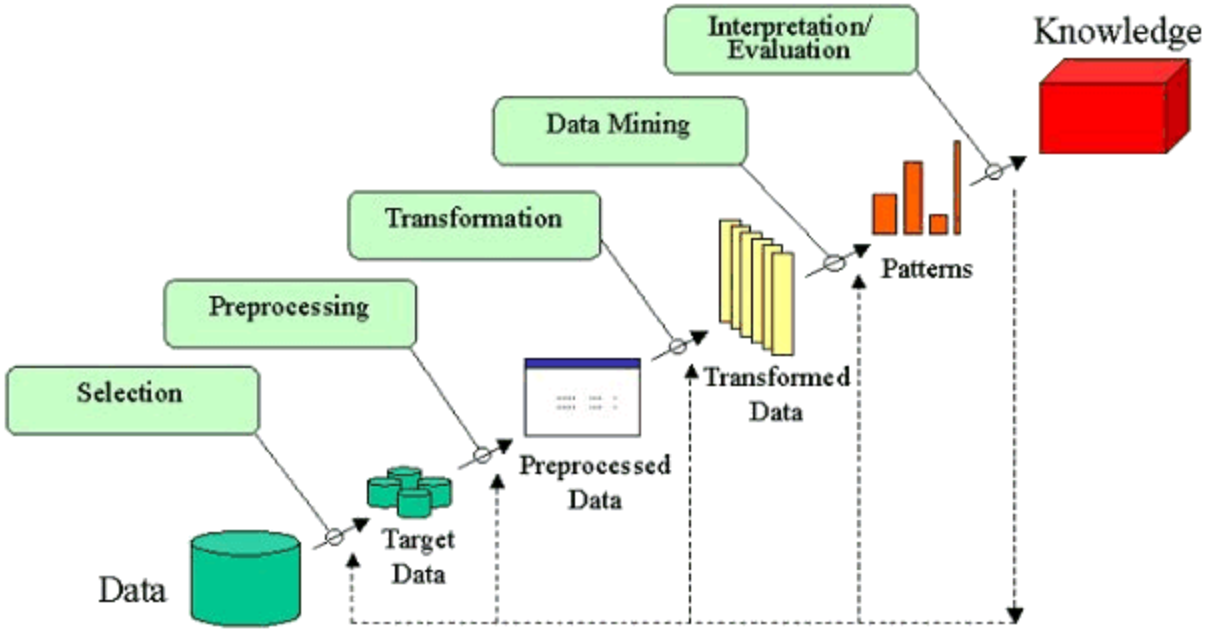
What is the KDD Process?

The term *Knowledge Discovery in Databases*, or KDD for short, refers to the broad process of finding knowledge in data, and emphasizes the "high-level" application of particular data mining methods. It is of interest to researchers in machine learning, pattern recognition, databases, statistics, artificial intelligence, knowledge acquisition for expert systems, and data visualization.

The unifying goal of the KDD process is to extract knowledge from data in the context of large databases.

It does this by using data mining methods (algorithms) to extract (identify) what is deemed knowledge, according to the specifications of measures and thresholds, using a database along with any required preprocessing, subsampling, and transformations of that database.

An Outline of the Steps of the KDD Process



The overall process of finding and interpreting patterns from data involves the repeated application of the following steps:

1. Developing an understanding of
    - the application domain
    - the relevant prior knowledge
    - the goals of the end-user
2. Creating a target data set: selecting a data set, or focusing on a subset of variables, or data samples, on which discovery is to be performed.
3. Data cleaning and preprocessing.
    - Removal of noise or outliers.
    - Collecting necessary information to model or account for noise.
    - Strategies for handling missing data fields.
    - Accounting for time sequence information and known changes.
4. Data reduction and projection.

    - Finding useful features to represent the data depending on the goal of the task.
    - Using dimensionality reduction or transformation methods to reduce the effective number of variables under consideration or to find invariant representations for the data.
5. Choosing the data mining task.
    - Deciding whether the goal of the KDD process is classification, regression, clustering, etc.
6. Choosing the data mining algorithm(s).
    - Selecting method(s) to be used for searching for patterns in the data.
    - Deciding which models and parameters may be appropriate.
    - Matching a particular data mining method with the overall criteria of the KDD process.
7. Data mining.
    - Searching for patterns of interest in a particular representational form or a set of such representations as classification rules or trees, regression, clustering, and so forth.
8. Interpreting mined patterns.
9. Consolidating discovered knowledge.

The terms *knowledge discovery* and *data mining* are distinct.

KDD refers to the overall process of discovering useful knowledge from data. It involves the evaluation and possibly interpretation of the patterns to make the decision of what qualifies as knowledge. It also includes the choice of encoding schemes, preprocessing, sampling, and projections of the data prior to the data mining step. **Data mining** refers to the application of algorithms for extracting patterns from data without the additional steps of the KDD process.

Definitions Related to the KDD Process

Knowledge discovery in databases is the non-trivial process of identifying valid, novel, potentially useful, and ultimately understandable patterns in data.

Data	A set of facts, F.
Pattern	An expression E in a language L describing facts in a subset F_E of F.
Process	KDD is a *multi-step process* involving data preparation, pattern searching, knowledge evaluation, and refinement with iteration after modification.

Valid	Discovered patterns should be true on new data with some degree of certainty. Generalize to the future (other data).
Novel	Patterns must be novel (should not be previously known).
Useful	Actionable; patterns should potentially lead to some useful actions.
Understandable	The process should lead to human insight. Patterns must be made understandable in order to facilitate a better understanding of the underlying data.

The Business Context of Data Mining

Why does an organisation have to practise data mining when it does not bring impact to their businesses? In product marketing, the marketing manager should identify the segment of the population who is most likely to respond to your product. Identifying these segments of population involves understanding the overall population and deploying the right technique to classify the population. Likewise, in predictive modelling, there are several ways to interact with the customers using different channels. These include direct marketing, print advertising, telemarketing, radio, television advertising and so on. It is only through data mining, that an analyst would conclude which is the optimal channel for sending the communication to the customers.

In addition to segmenting and targeting, data mining is also popularly used for budgeting the marketing spend, so the budget allocation can be optimised across marketing drivers. The analysis is carried out based on previous year spend and their impact on the sales. Therefore with the spend information for each driver, like, Print, TV, Radio, Online, etc, one could determine the ROIs for each driver that would uncover the impact of these channels on the sales. Based on this analysis the marketing manager could allocate media pend in the coming year to achieve the most effective results on sales.

Process improvement through data mining

The role of data in manufacturing has always been understated or unstated. The way companies cope with quality improvement has been transformed by new forms of data use and data analytics. The experts in the field report a considerable shift from exclusive dependence on post-manufacturing inspection work and retrospective analysis to the prediction and early identification of problem areas and maintenance requirements. New sources of data—from sensors to callcenter conversations—are bringing traditional product inspections on a new level. By transforming the management of quality and safety in asset-based businesses, these innovations are gradually improving manufacturing sector.

Data transforms technology, and it's only the beginning of striking changes.

The quality and safety revolution in organizations was marked by numerous **technical breakthroughs** such as real-time data from connected vehicle sensors and GPS and text derived from warranty reports and conversions of callcenter speech conversations, just to name a few. On the other hand, the data is now combined in a repository that allows for multiple data formats and analysis across them.This is where exactly machine learning algorithms come to play. Their role is to identify trends in the data and to make predictions.

Why to use data mining?

Businesses use data mining to draw conclusions and solve specific problems. One of the key benefits of data mining is that it is fundamentally **applicable to any process and helps improve the flexibility and efficiency of operations.** Thus, data use in manufacturing facilitates schedule adherence, monitoring automation, modeling for capacity, and reduction of waste. The departments are completely transformed and factories become smarter by achieving full data transparency.

How manufacturing businesses take advantage of data mining

ABB, a huge manufacturer of a global importance, is currently using **process mining** for purchase-to-pay and production processes. Earlier, the employees from the *ABB* plant in Hanau, Germany, would extract evaluations from their SAP systems several times a day, import them into Excel, and use complex formulas to analyze and understand processes. Today, the relevant production and assembly team leaders at *ABB* receive an email in the morning that outlines the previous day's production variants, throughput times, and number of rejections. As a result, the plant's full ecosystem of quality improvement processes is immediately visible with process mining. The system only gets better at identifying patterns as more data gets fed in. Instead of relying on complex manual analysis of processes, operational processes provide instant results.

Drastic changes have impacted **vehicle manufacturing industry** too. In this sector, the products are relatively expensive, with high-end manufacturers focusing on service and product quality. They note that the business benefits related to the i**ntroduction of data-driven innovations** have all the chances to speed up identification and resolution of quality problems, as well as cut warranty spending, which amounts to between 2-6 % of total sales in the automobile industry. For the

customers and users of these vehicles and machines, early identification and preventive maintenance often results in greater uptime. For instance, in one case involving an automotive company, 28,000 vehicles were saved from recall by the identification of a problem before vehicles hit the market.

Data mining tools can be very beneficial for discovering interesting and useful patterns in complicated manufacturing quality improvement processes. These patterns can be used to improve manufacturing quality. However, data accumulated in manufacturing plants have unique characteristics, such as unbalanced distribution of the target attribute, and a small training set relative to the number of input features. Anyways, business process improvement has to start somewhere. Using an approach that incorporates big data, analytics and business intelligence approach is simply the most reliable, proven way to make improvements that last. Once you know what to measure, track it, analyse it, and improve it, you'll have the right foundations in place to enhance processes throughout your business. Time and product waste will be the things of the past.

Data mining as a tool for research and knowledge development in nursing.

The ability to collect and store data has grown at a dramatic rate in all disciplines over the past two decades. Healthcare has been no exception. The shift toward evidence-based practice and outcomes research presents significant opportunities and challenges to extract meaningful information from massive amounts of clinical data to transform it into the best available knowledge to guide nursing practice. Data mining, a step in the process of Knowledge Discovery in Databases, is a method of unearthing information from large data sets. Built upon statistical analysis, artificial intelligence, and machine learning technologies, data mining can analyze massive amounts of data and provide useful and interesting information about patterns and relationships that exist within the data that might otherwise be missed. As domain experts, nurse researchers are in ideal positions to use this proven technology to transform the information that is available in existing data repositories into useful and understandable knowledge to guide nursing practice and for active interdisciplinary collaboration and research.

Data mining in marketing

• Data mining technology allows to learn more about their customers and make smart marketing decisions.

• The data mining business, grows 10 percent a year as the amount of data produced is booming.

• DM Information can help to
  – increase return on investment (ROI)
  – improve CRM and market analysis
  – reduce marketing campaign costs

– facilitate fraud detection and customer retention.

• The 4Ps is one way of the best way of defining the marketing:
  –Product (or Service)
  –Price
  –Place
  –Promotion

Benefits Using Data Mining in Marketing

• Predict future trends
• customer purchase habits
• Help with decision making
• Improve company revenue and lower costs
• Market basket analysis
• Quick Fraud detection

Barriers Using Data Mining in Marketing

• User privacy/security
• Amount of data is overwhelming
• Great cost at implementation stage
• Possible misuse of information
• Possible in accuracy of data

Data Mining Techniques for Marketing

• Knowledge-based Marketing
• Market Basket Analysis
• Social Media Marketing

Knowledge-based Marketing

• It is marketing which makes use of the macro- and micro-environmental knowledge that is available to the marketing functional unit in an organization.
• There are three major areas of application of data mining for knowledge-based marketing are customers profiling, deviation analysis, and trend analysis.
• The Customers profiling systems can analyse the frequency of purchases, companies can know how many times the customers can buy this product or visit the store.
• The Deviation analysis gives the marketer a good capability to query changes that occurred as a result of recent price changes or promotions.
• The Trend analysis can determine trends in sales, costs and profits by products or markets in order to achieve the highest amount of sales.

Market Basket Analysis

• Most common and useful types of data analysis for marketing and retailing.
• Determine what products customers purchase together.

• Improve the effectiveness of marketing and sales tactics using customer data already available to the company.

Social Media Marketing

• SMM is a form of internet marketing that implements various social media networks in order to achieve marketing communication and branding goals.
• SMM primarily covers activities involving social sharing of content, videos, and images for marketing purposes, as well as paid social media advertising.

Data Mining Tools for Marketing

• WEKA
• Rapid Miner
• R-Programming Tool
• Python Based Orange and NTLK
• KNIME

Major Data Mining Techniques: Classification and Prediction

There are two forms of data analysis that can be used for extracting models describing important classes or to predict future data trends. These two forms are as follows −

- Classification
- Prediction

Classification models predict categorical class labels; and prediction models predict continuous valued functions. For example, we can build a classification model to categorize bank loan applications as either safe or risky, or a prediction model to predict the expenditures in dollars of potential customers on computer equipment given their income and occupation.

What is classification?

Following are the examples of cases where the data analysis task is Classification −

- A bank loan officer wants to analyze the data in order to know which customer (loan applicant) are risky or which are safe.
- A marketing manager at a company needs to analyze a customer with a given profile, who will buy a new computer.

In both of the above examples, a model or classifier is constructed to predict the categorical labels. These labels are risky or safe for loan application data and yes or no for marketing data.

What is prediction?

Following are the examples of cases where the data analysis task is Prediction −

Suppose the marketing manager needs to predict how much a given customer will spend during a sale at his company. In this example we are bothered to predict a numeric value. Therefore the data analysis task is an example of numeric prediction. In this case, a model or a predictor will be constructed that predicts a continuous-valued-function or ordered value.

Note − Regression analysis is a statistical methodology that is most often used for numeric prediction.

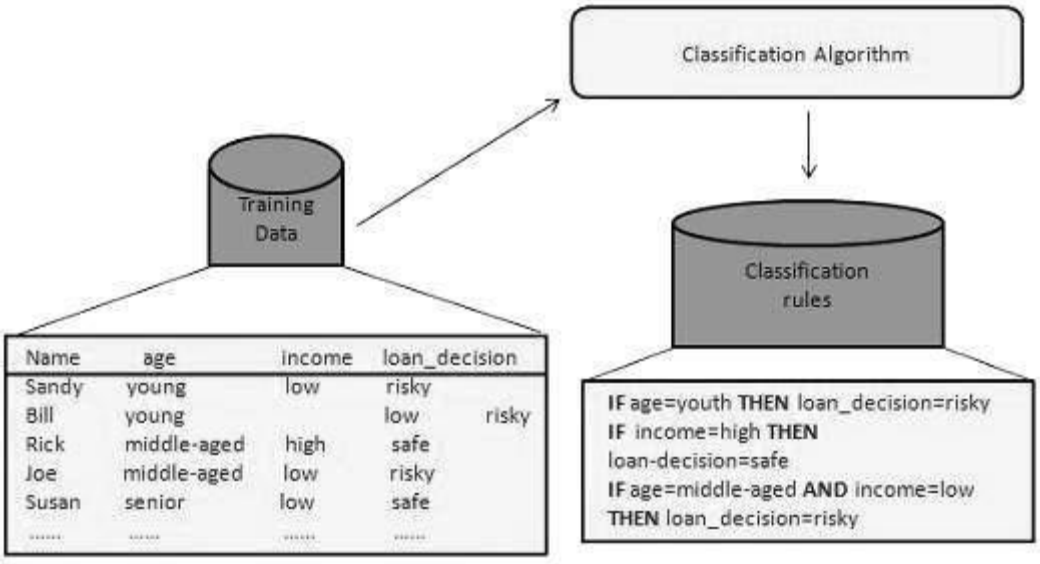
How Does Classification Works?

With the help of the bank loan application that we have discussed above, let us understand the working of classification. The Data Classification process includes two steps −

- Building the Classifier or Model
- Using Classifier for Classification

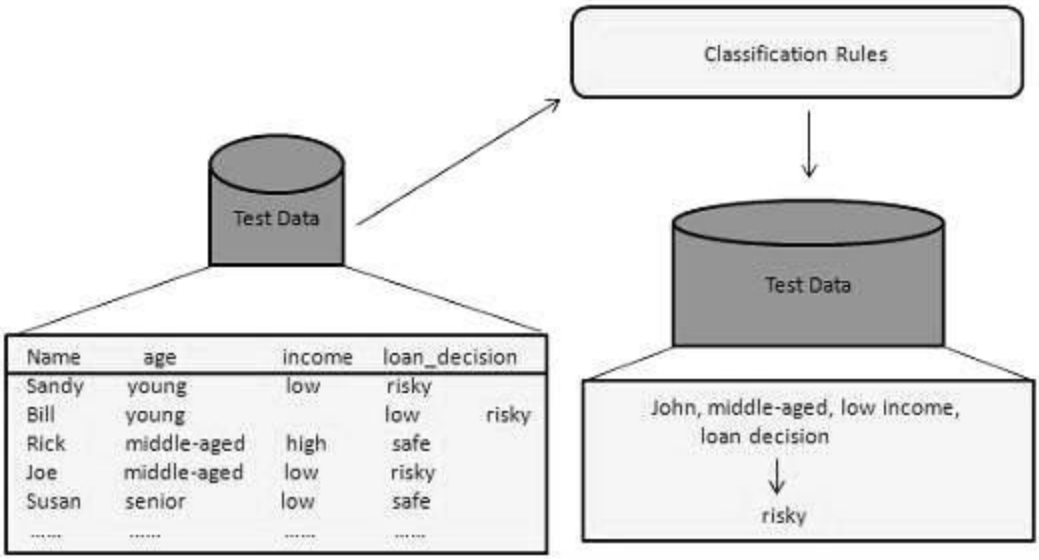
Building the Classifier or Model

- This step is the learning step or the learning phase.
- In this step the classification algorithms build the classifier.
- The classifier is built from the training set made up of database tuples and their associated class labels.
- Each tuple that constitutes the training set is referred to as a category or class. These tuples can also be referred to as sample, object or data points.



Using Classifier for Classification

In this step, the classifier is used for classification. Here the test data is used to estimate the accuracy of classification rules. The classification rules can be applied to the new data tuples if the accuracy is considered acceptable.



Classification and Prediction Issues

The major issue is preparing the data for Classification and Prediction. Preparing the data involves the following activities −

- **Data Cleaning** − Data cleaning involves removing the noise and treatment of missing values. The noise is removed by applying smoothing techniques and the problem of missing values is solved by replacing a missing value with most commonly occurring value for that attribute.
- **Relevance Analysis** − Database may also have the irrelevant attributes. Correlation analysis is used to know whether any two given attributes are related.
- **Data Transformation and reduction** − The data can be transformed by any of the following methods.
    - **Normalization** − The data is transformed using normalization. Normalization involves scaling all values for given attribute in order to make them fall within a small specified range. Normalization is used when in the learning step, the neural networks or the methods involving measurements are used.
    - **Generalization** − The data can also be transformed by generalizing it to the higher concept. For this purpose we can use the concept hierarchies.

Note − Data can also be reduced by some other methods such as wavelet transformation, binning, histogram analysis, and clustering.

Comparison of Classification and Prediction Methods

Here is the criteria for comparing the methods of Classification and Prediction −

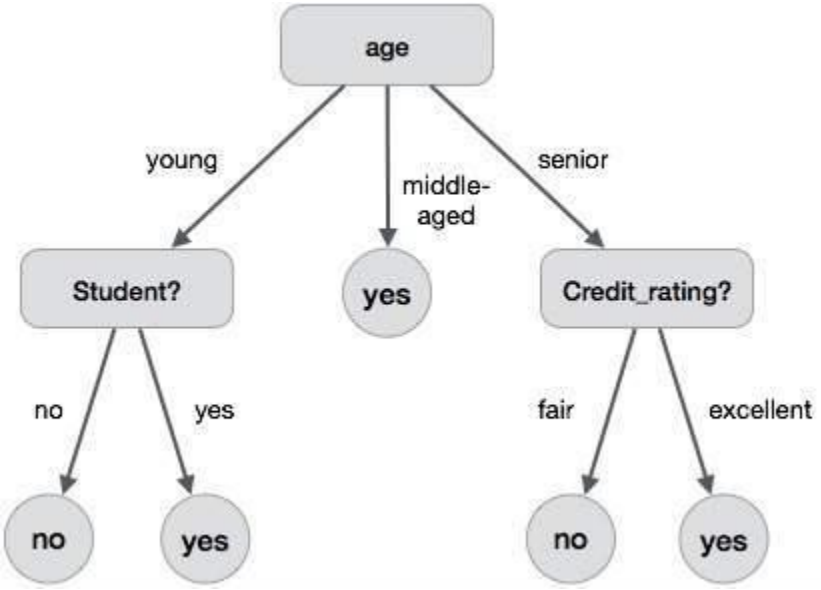
- **Accuracy** − Accuracy of classifier refers to the ability of classifier. It predict the class label correctly and the accuracy of the predictor refers to how well a given predictor can guess the value of predicted attribute for a new data.

- **Speed** − This refers to the computational cost in generating and using the classifier or predictor.
- **Robustness** − It refers to the ability of classifier or predictor to make correct predictions from given noisy data.
- **Scalability** − Scalability refers to the ability to construct the classifier or predictor efficiently; given large amount of data.
- **Interpretability** − It refers to what extent the classifier or predictor understands.

Classification by Decision Tree Induction

A decision tree is a structure that includes a root node, branches, and leaf nodes. Each internal node denotes a test on an attribute, each branch denotes the outcome of a test, and each leaf node holds a class label. The topmost node in the tree is the root node.

The following decision tree is for the concept buy_computer that indicates whether a customer at a company is likely to buy a computer or not. Each internal node represents a test on an attribute. Each leaf node represents a class.



The benefits of having a decision tree are as follows −

- It does not require any domain knowledge.
- It is easy to comprehend.
- The learning and classification steps of a decision tree are simple and fast.

Decision Tree Induction Algorithm

A machine researcher named J. Ross Quinlan in 1980 developed a decision tree algorithm known as ID3 (Iterative Dichotomiser). Later, he presented C4.5, which was

the successor of ID3. ID3 and C4.5 adopt a greedy approach. In this algorithm, there is no backtracking; the trees are constructed in a top-down recursive divide-and-conquer manner.

Generating a decision tree form training tuples of data partition D

Algorithm : Generate_decision_tree

Input:

Data partition, D, which is a set of training tuples
and their associated class labels.
attribute_list, the set of candidate attributes.
Attribute selection method, a procedure to determine the
splitting criterion that best partitions that the data
tuples into individual classes. This criterion includes a
splitting_attribute and either a splitting point or splitting
subset.

Output:

A Decision Tree

Method

```
create a node N;

if tuples in D are all of the same class, C then
   return N as leaf node labeled with class C;

if attribute_list is empty then
   return N as leaf node with labeled
   with majority class in D;|| majority voting

apply attribute_selection_method(D, attribute_list)
to find the best splitting_criterion;
label node N with splitting_criterion;

if splitting_attribute is discrete-valued and
   multiway splits allowed then  // no restricted to binary trees

attribute_list = splitting attribute; // remove splitting attribute
for each outcome j of splitting criterion

   // partition the tuples and grow subtrees for each partition
   let Dj be the set of data tuples in D satisfying outcome j; // a
partition

   if Dj is empty then
      attach a leaf labeled with the majority
      class in D to node N;
   else
      attach the node returned by Generate
      decision tree(Dj, attribute list) to node N;
```

```
   end for
return N;
```

Tree Pruning

Tree pruning is performed in order to remove anomalies in the training data due to noise or outliers. The pruned trees are smaller and less complex.

Tree Pruning Approaches

There are two approaches to prune a tree −

- **Pre-pruning** − The tree is pruned by halting its construction early.
- **Post-pruning** - This approach removes a sub-tree from a fully grown tree.

Cost Complexity

The cost complexity is measured by the following two parameters −

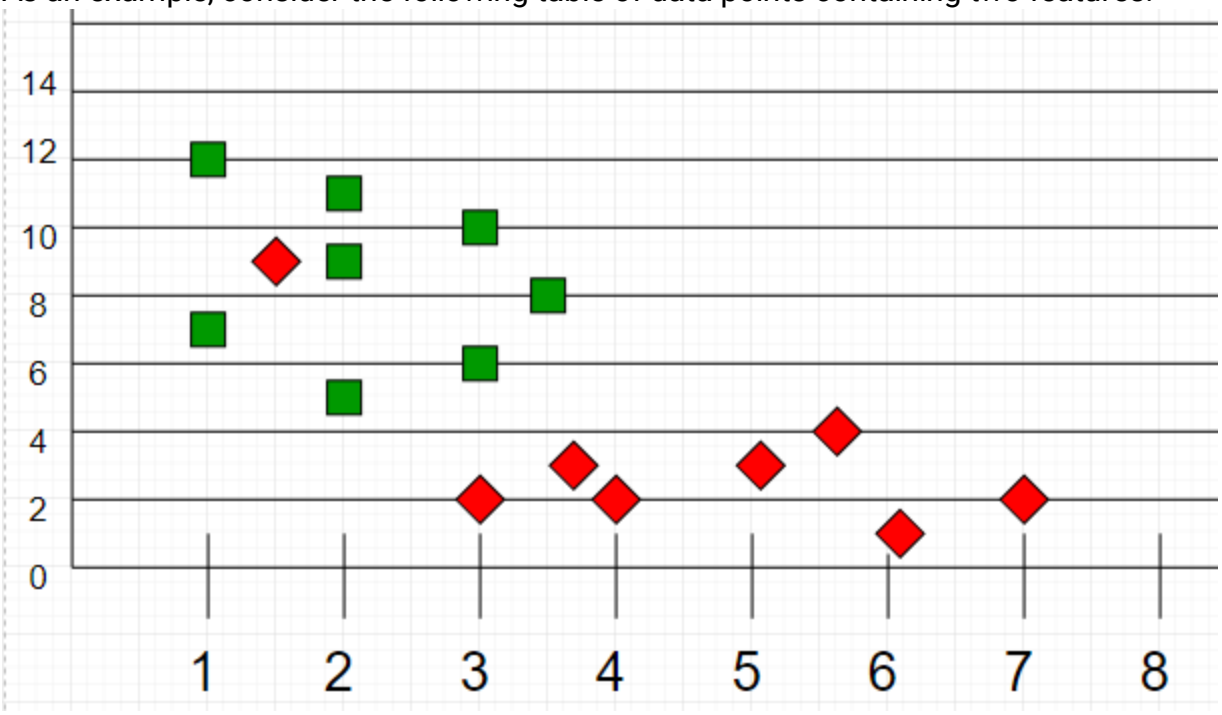
- Number of leaves in the tree, and
- Error rate of the tree.

KNN Algorithm

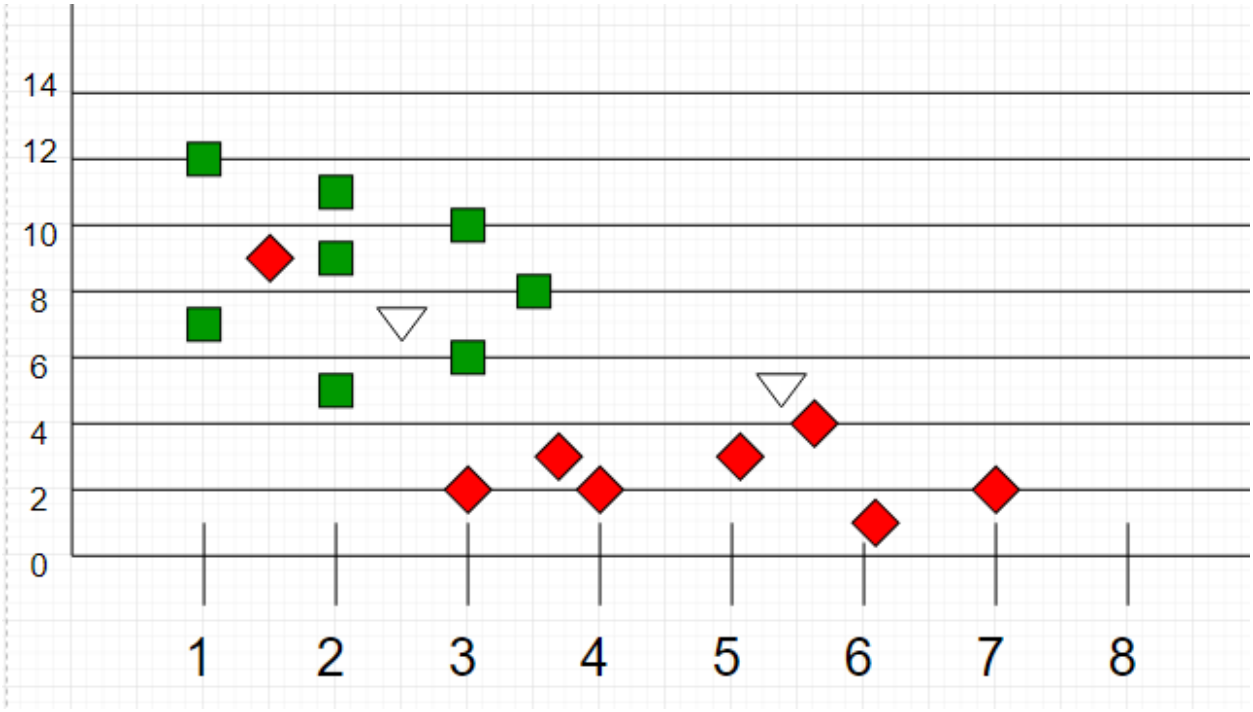
K-Nearest Neighbors is one of the most basic yet essential classification algorithms in Machine Learning. It belongs to the supervised learning domain and finds intense application in pattern recognition, data mining and intrusion detection.

It is widely disposable in real-life scenarios since it is non-parametric, meaning, it does not make any underlying assumptions about the distribution of data (as opposed to other algorithms such as GMM, which assume a Gaussian distribution of the given data).
We are given some prior data (also called training data), which classifies coordinates into groups identified by an attribute.

As an example, consider the following table of data points containing two features:



Now, given another set of data points (also called testing data), allocate these points a group by analyzing the training set. Note that the unclassified points are marked as 'White'.



Intuition

If we plot these points on a graph, we may be able to locate some clusters or groups.

Now, given an unclassified point, we can assign it to a group by observing what group its nearest neighbors belong to. This means a point close to a cluster of points classified as 'Red' has a higher probability of getting classified as 'Red'.
Intuitively, we can see that the first point (2.5, 7) should be classified as 'Green' and the second point (5.5, 4.5) should be classified as 'Red'.

Algorithm
Let m be the number of training data samples. Let p be an unknown point.

1. Store the training samples in an array of data points arr[]. This means each element of this array represents a tuple (x, y).

```
2. for i=0 to m:

3.   Calculate Euclidean distance d(arr[i], p).
```

4. Make set S of K smallest distances obtained. Each of these distances corresponds to an already classified data point.
5. Return the majority label among S.

K can be kept as an odd number so that we can calculate a clear majority in the case where only two groups are possible (e.g. Red/Blue). With increasing K, we get smoother, more defined boundaries across different classifications. Also, the accuracy of the above classifier increases as we increase the number of data points in the training set.

UNIT 4

Cluster detection

Cluster is a group of objects that belongs to the same class. In other words, similar objects are grouped in one cluster and dissimilar objects are grouped in another cluster.

What is Clustering?

Clustering is the process of making a group of abstract objects into classes of similar objects.

Points to Remember

- A cluster of data objects can be treated as one group.
- While doing cluster analysis, we first partition the set of data into groups based on data similarity and then assign the labels to the groups.
- The main advantage of clustering over classification is that, it is adaptable to changes and helps single out useful features that distinguish different groups.

Applications of Cluster Analysis

- Clustering analysis is broadly used in many applications such as market research, pattern recognition, data analysis, and image processing.
- Clustering can also help marketers discover distinct groups in their customer base. And they can characterize their customer groups based on the purchasing patterns.
- In the field of biology, it can be used to derive plant and animal taxonomies, categorize genes with similar functionalities and gain insight into structures inherent to populations.
- Clustering also helps in identification of areas of similar land use in an earth observation database. It also helps in the identification of groups of houses in a city according to house type, value, and geographic location.
- Clustering also helps in classifying documents on the web for information discovery.
- Clustering is also used in outlier detection applications such as detection of credit card fraud.
- As a data mining function, cluster analysis serves as a tool to gain insight into the distribution of data to observe characteristics of each cluster.

Requirements of Clustering in Data Mining

The following points throw light on why clustering is required in data mining −

- **Scalability** − We need highly scalable clustering algorithms to deal with large databases.
- **Ability to deal with different kinds of attributes** − Algorithms should be capable to be applied on any kind of data such as interval-based (numerical) data, categorical, and binary data.

- **Discovery of clusters with attribute shape** − The clustering algorithm should be capable of detecting clusters of arbitrary shape. They should not be bounded to only distance measures that tend to find spherical cluster of small sizes.
- **High dimensionality** − The clustering algorithm should not only be able to handle low-dimensional data but also the high dimensional space.
- **Ability to deal with noisy data** − Databases contain noisy, missing or erroneous data. Some algorithms are sensitive to such data and may lead to poor quality clusters.
- **Interpretability** − The clustering results should be interpretable, comprehensible, and usable.

Clustering Methods

Clustering methods can be classified into the following categories −

- Partitioning Method
- Hierarchical Method
- Density-based Method
- Grid-Based Method
- Model-Based Method
- Constraint-based Method

Partitioning Method

Suppose we are given a database of ‘n’ objects and the partitioning method constructs ‘k’ partition of data. Each partition will represent a cluster and k ≤ n. It means that it will classify the data into k groups, which satisfy the following requirements −

- Each group contains at least one object.
- Each object must belong to exactly one group.

Points to remember −

- For a given number of partitions (say k), the partitioning method will create an initial partitioning.
- Then it uses the iterative relocation technique to improve the partitioning by moving objects from one group to other.

Hierarchical Methods

This method creates a hierarchical decomposition of the given set of data objects. We can classify hierarchical methods on the basis of how the hierarchical decomposition is formed. There are two approaches here −

- Agglomerative Approach
- Divisive Approach

Agglomerative Approach

This approach is also known as the bottom-up approach. In this, we start with each object forming a separate group. It keeps on merging the objects or groups that are close to one another. It keep on doing so until all of the groups are merged into one or until the termination condition holds.

Divisive Approach

This approach is also known as the top-down approach. In this, we start with all of the objects in the same cluster. In the continuous iteration, a cluster is split up into smaller clusters. It is down until each object in one cluster or the termination condition holds. This method is rigid, i.e., once a merging or splitting is done, it can never be undone.

Approaches to Improve Quality of Hierarchical Clustering

Here are the two approaches that are used to improve the quality of hierarchical clustering −

- Perform careful analysis of object linkages at each hierarchical partitioning.
- Integrate hierarchical agglomeration by first using a hierarchical agglomerative algorithm to group objects into micro-clusters, and then performing macro-clustering on the micro-clusters.

Density-based Method

This method is based on the notion of density. The basic idea is to continue growing the given cluster as long as the density in the neighborhood exceeds some threshold, i.e., for each data point within a given cluster, the radius of a given cluster has to contain at least a minimum number of points.

Grid-based Method

In this, the objects together form a grid. The object space is quantized into finite number of cells that form a grid structure.

Advantages

- The major advantage of this method is fast processing time.
- It is dependent only on the number of cells in each dimension in the quantized space.

Model-based methods

In this method, a model is hypothesized for each cluster to find the best fit of data for a given model. This method locates the clusters by clustering the density function. It reflects spatial distribution of the data points.

This method also provides a way to automatically determine the number of clusters based on standard statistics, taking outlier or noise into account. It therefore yields robust clustering methods.

Constraint-based Method

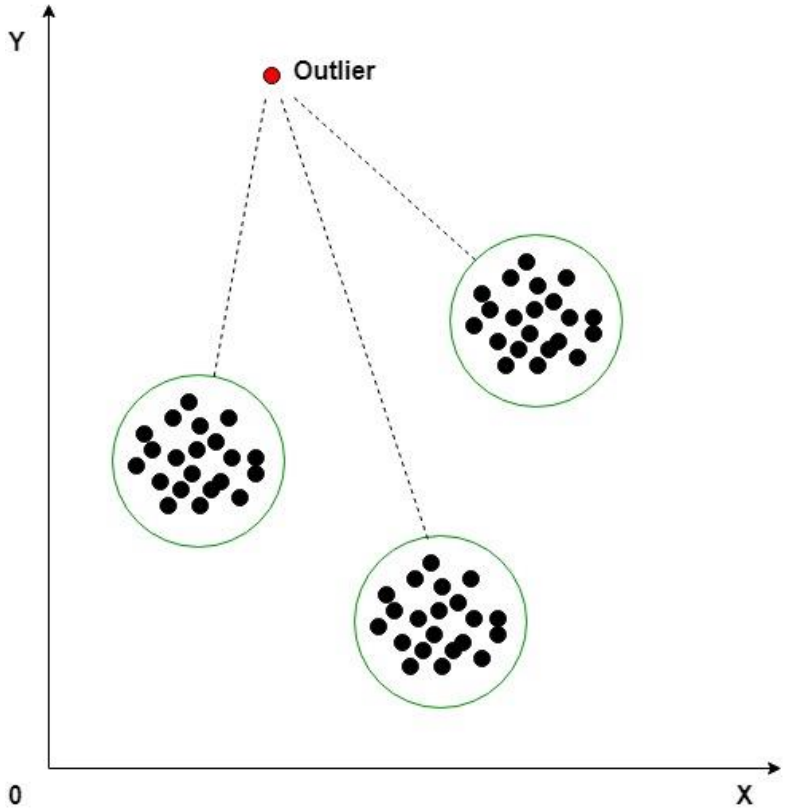
In this method, the clustering is performed by the incorporation of user or application-oriented constraints. A constraint refers to the user expectation or the properties of desired clustering results. Constraints provide us with an interactive way of communication with the clustering process. Constraints can be specified by the user or the application requirement.

Outlier Analysis

An **outlier** is an object that deviates significantly from the rest of the objects. They can be caused by measurement or execution error. The analysis of outlier data is referred to as outlier analysis or outlier mining.

Why outlier analysis?

Most data mining methods discard outliers noise or exceptions, however, in some applications such as fraud detection, the rare events can be more interesting than the more regularly occurring one and hence, the outlier analysis becomes important in such case.



Detecting Outlier

Clustering based outlier detection using distance to the closest cluster:

In K-Means clustering technique, each cluster has a mean value. Objects belong to the cluster whose mean value is closest to it. In order to identify the Outlier, firstly we need to initialize the threshold value such that any distance of any data point greater than it from its nearest cluster identifies it as an outlier for our purpose. Then we need to find the distance of the test data to each cluster mean. Now, if the distance between the test data and the closest cluster to it is greater than the threshold value then we will classify the test data as an outlier.

Algorithm:

1. Calculate the mean of each cluster
2. Initialize the Threshold value
3. Calculate the distance of the test data from each cluster mean
4. Find the nearest cluster to the test data
5. if (Distance>Threshold) then, Outlier

Memory-Based Reasoning

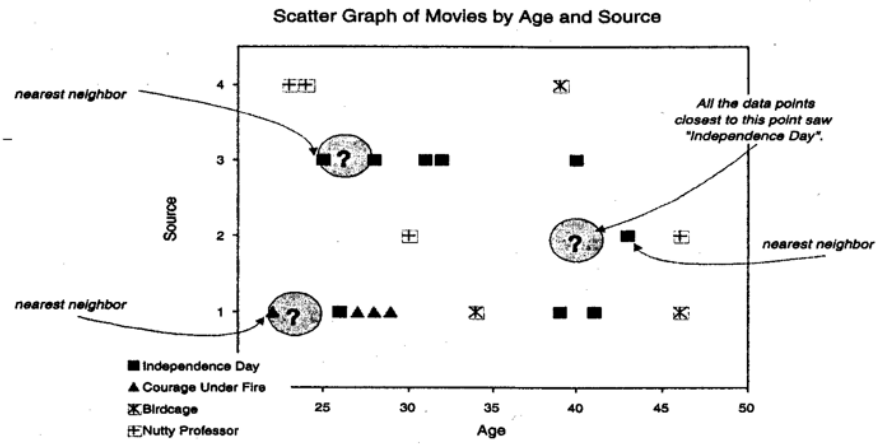
- Memory-Based Reasoning(MBR) is
  - Identifying similar cases from experience
  - Applying the information from these cases to the problem at hand.
  - MBR finds neighbors similar to a new record and uses the neighbors for classification and prediction.

- It cares about the existence of two operations
  - Distance function ; assigns a distance between any two records
  - Combination function ; combines the results from the neighbors to arrive at an answer.

- Applications of MBR span many areas;
  - Fraud detection
  - Customer response prediction
  - Medical treatments
  - Classifying responses

How does MBR work?

- What is the most likely movie last seen by a respondent based on the source of the record and the age of the individual?



- MBR has two distinct phases
  - The learning phase generates the historical database
  - The prediction phase applies MBR to new cases

The three main issues in solving a problem with MBR

- Choosing the appropriate set of historical records
  - The historical records, also known as the training set, is a subset of available records.
  - The training set needs to provide good coverage of the records so that the nearest neighbors to an unknown record are useful for predictive purposes.

- Representing the historical records
  - The performance of MBR in making predictions depends on how the training set is represented in the computer.

- Determining the distance function, Combination function, and number of neighbors
  - The distance function, combination function, and number of neighbors are the key ingredients in determining how good MBR is at producing results.

- Strengths of Memory-Based Reasoning
  - It produces results that are readily understandable.
  - It is applicable to arbitrary data types, even non-relational data.
  - It works efficiently on almost any number of fields.
  - Maintaining the training set requires a minimal amount of effort.

- Weaknesses of Memory-Based Reasoning
  - It is computationally expensive when doing classification and prediction.
  - It requires a large amount of storage for the training set.
  - Results can be dependent on the choice of distance function, combination function, and number of neighbors

Link Analysis

In network theory, **link analysis** is a data-analysis technique used to evaluate relationships (connections) between nodes. Relationships may be identified among various types of nodes (objects), including organizations, people and transactions. Link analysis has been used for investigation of criminal activity (fraud detection, counterterrorism, and intelligence), computer security analysis, search engine optimization, market research, medical research, and art.

Knowledge discovery

Knowledge discovery is an iterative and interactive process used to identify, analyze and visualize patterns in data. Network analysis, link analysis and social network analysis are all methods of knowledge discovery, each a corresponding subset of the prior method. Most knowledge discovery methods follow these steps (at the highest level):

1. Data processing
2. Transformation
3. Analysis
4. Visualization

Data gathering and processing requires access to data and has several inherent issues, including information overload and data errors. Once data is collected, it will need to be transformed into a format that can be effectively used by both human and computer analyzers. Manual or computer-generated visualizations tools may be mapped from the data, including network charts. Several algorithms exist to help with analysis of data – Dijkstra's algorithm, breadth-first search, and depth-first search.

Link analysis focuses on analysis of relationships among nodes through visualization methods (network charts, association matrix). Here is an example of the relationships that may be mapped for crime investigations:

Relationship/Network	Data Sources
1. Trust	Prior contacts in family, neighborhood, school, military, club or organization. Public and court records. Data may only be available in suspect's native country.
2. Task	Logs and records of phone calls, electronic mail, chat rooms, instant messages, Web site visits. Travel records. Human intelligence: observation of meetings and attendance at common events.
3. Money & Resources	Bank account and money transfer records. Pattern and location of credit card use. Prior court records. Human intelligence: observation of visits to alternate banking resources such as Hawala.
4. Strategy & Goals	Web sites. Videos and encrypted disks delivered by courier. Travel records. Human intelligence: observation of meetings and attendance at common events.

Link analysis is used for 3 primary purposes:

1. Find matches in data for known patterns of interest;
2. Find anomalies where known patterns are violated;
3. Discover new patterns of interest (social network analysis, data mining).

Applications

- FBI Violent Criminal Apprehension Program (ViCAP)
- Iowa State Sex Crimes Analysis System
- Minnesota State Sex Crimes Analysis System (MIN/SCAP)
- Washington State Homicide Investigation Tracking System (HITS)
- New York State Homicide Investigation & Lead Tracking (HALT)
- New Jersey Homicide Evaluation & Assessment Tracking (HEAT)
- Pennsylvania State ATAC Program.
- Violent Crime Linkage Analysis System (ViCLAS)

Issues with link analysis

Information overload

With the vast amounts of data and information that are stored electronically, users are confronted with multiple unrelated sources of information available for analysis. Data analysis techniques are required to make effective and efficient use of the data. Palshikar classifies data analysis techniques

into two categories – (statistical models, time-series analysis, clustering and classification, matching algorithms to detect anomalies) and artificial intelligence (AI) techniques (data mining, expert systems, pattern recognition, machine learning techniques, neural networks).

Bolton & Hand define statistical data analysis as either supervised or unsupervised methods. Supervised learning methods require that rules are defined within the system to establish what is expected or unexpected behavior. Unsupervised learning methods review data in comparison to the norm and detect statistical outliers. Supervised learning methods are limited in the scenarios that can be handled as this method requires that training rules are established based on previous patterns. Unsupervised learning methods can provide detection of broader issues, however, may result in a higher false-positive ratio if the behavioral norm is not well established or understood.

Data itself has inherent issues including integrity (or lack of) and continuous changes. Data may contain "errors of omission and commission because of faulty collection or handling, and when entities are actively attempting to deceive and/or conceal their actions". Sparrow highlights incompleteness (inevitability of missing data or links), fuzzy boundaries (subjectivity in deciding what to include) and dynamic changes (recognition that data is ever-changing) as the three primary problems with data analysis.

Once data is transformed into a usable format, open texture and cross referencing issues may arise. Open texture was defined by Waismann as the unavoidable uncertainty in meaning when empirical terms are used in different contexts. Uncertainty in meaning of terms presents problems when attempting to search and cross reference data from multiple sources.[18]

The primary method for resolving data analysis issues is reliance on domain knowledge from an expert. This is a very time-consuming and costly method of conducting link analysis and has inherent problems of its own. McGrath et al. conclude that the layout and presentation of a network diagram have a significant impact on the user's "perceptions of the existence of groups in networks".[19] Even using domain experts may result in differing conclusions as analysis may be subjective.

Prosecution vs. crime prevention

Link analysis techniques have primarily been used for prosecution, as it is far easier to review historical data for patterns than it is to attempt to predict future actions.

Krebs demonstrated the use of an association matrix and link chart of the terrorist network associated with the 19 hijackers responsible for the September 11th attacks by mapping publicly available details made available following the attacks.[3] Even with the advantages of hindsight and publicly available information on people, places and transactions, it is clear that there is missing data.

Alternatively, Picarelli argued that use of link analysis techniques could have been used to identify and potentially prevent illicit activities within the Aum Shinrikyo network. "We must be careful of 'guilt by association'. Being linked to a terrorist does not prove guilt – but it does invite investigation." Balancing the legal concepts of probable cause, right to privacy and freedom of association become challenging when reviewing potentially sensitive data with the objective to prevent crime or illegal activity that has not yet occurred.

Proposed solutions

There are four categories of proposed link analysis solutions:[21]

1. Heuristic-based
2. Template-based

3. Similarity-based
4. Statistical

Heuristic-based tools utilize decision rules that are distilled from expert knowledge using structured data. Template-based tools employ Natural Language Processing (NLP) to extract details from unstructured data that are matched to pre-defined templates. Similarity-based approaches use weighted scoring to compare attributes and identify potential links. Statistical approaches identify potential links based on lexical statistics.

Association Rule Mining

Association rule mining finds interesting associations and relationships among large sets of data items. This rule shows how frequently a itemset occurs in a transaction. A typical example is Market Based Analysis.

Market Based Analysis is one of the key techniques used by large relations to show associations between items.It allows retailers to identify relationships between the items that people buy together frequently.

Given a set of transactions, we can find rules that will predict the occurrence of an item based on the occurrences of other items in the transaction.

TID	ITEMS
1	Bread, Milk
2	Bread, Diaper, Beer, Eggs
3	Milk, Diaper, Beer, Coke
4	Bread, Milk, Diaper, Beer
5	Bread, Milk, Diaper, Coke

Before we start defining the rule, let us first see the basic definitions.

Support Count( ) – Frequency of occurrence of a itemset.

```
Here   ({Milk, Bread, Diaper})=2
```

Frequent Itemset – An itemset whose support is greater than or equal to minsup threshold.

Association Rule – An implication expression of the form X -> Y, where X and Y are any 2 itemsets.

```
Example: {Milk, Diaper}->{Beer}
```

Rule Evaluation Metrics –

- **Support(s) –**
  The number of transactions that include items in the {X} and {Y} parts of the rule as a percentage of the total number of transaction.It is a measure of how frequently the collection of items occur together as a percentage of all transactions.
- **Support = (X+Y) total –**
  It is interpreted as fraction of transactions that contain both X and Y.
- **Confidence(c) –**
  It is the ratio of the no of transactions that includes all items in {B} as well as the no of transactions that includes all items in {A} to the no of transactions that includes all items in {A}.
- **Conf(X=>Y) = Supp(X Y) Supp(X) –**
  It measures how often each item in Y appears in transactions that contains items in X also.
- **Lift(l) –**
  The lift of the rule X=>Y is the confidence of the rule divided by the expected confidence, assuming that the itemsets X and Y are independent of each other.The expected confidence is the confidence divided by the frequency of {Y}.
- **Lift(X=>Y) = Conf(X=>Y) Supp(Y) –**
  Lift value near 1 indicates X and Y almost often appear together as expected, greater than 1 means they appear together more than expected and less than 1 means they appear less than expected.Greater lift values indicate stronger association.

Genetic Algorithms

Genetic Algorithms(GAs) are adaptive heuristic search algorithms that belong to the larger part of evolutionary algorithms. Genetic algorithms are based on the ideas of natural selection and genetics. These are intelligent exploitation of random search provided with historical data to direct the search into the region of better performance in solution space. **They are commonly used to generate high-quality solutions for optimization problems and search problems.**

Genetic algorithms simulate the process of natural selection which means those species who can adapt to changes in their environment are able to survive and reproduce and go to next generation. In simple words, they simulate “survival of the fittest” among individual of consecutive generation for solving a problem. **Each generation consist of a population of individuals** and each individual represents a point in search space and possible solution. Each individual is represented as a string of character/integer/float/bits. This string is analogous to the Chromosome.

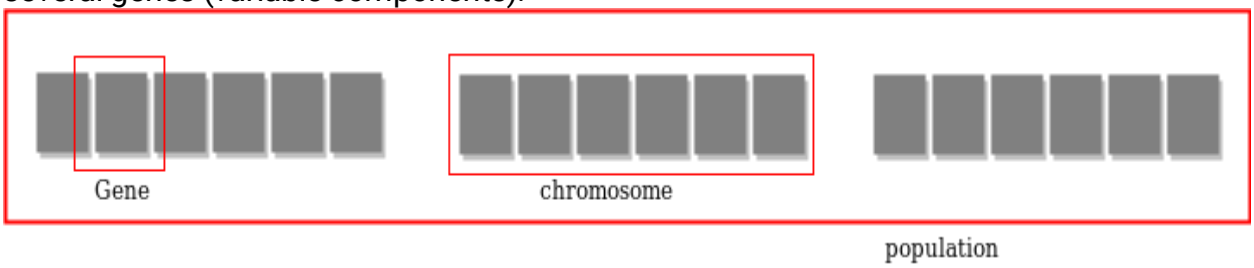
Foundation of Genetic Algorithms

Genetic algorithms are based on an analogy with genetic structure and behavior of chromosome of the population. Following is the foundation of GAs based on this analogy –

1. Individual in population compete for resources and mate
2. Those individuals who are successful (fittest) then mate to create more offspring than others
3. Genes from "fittest" parent propagate throughout the generation, that is sometimes parents create offspring which is better than either parent.
4. Thus each successive generation is more suited for their environment.

Search space

The population of individuals are maintained within search space. Each individual represent a solution in search space for given problem. Each individual is coded as a finite length vector (analogous to chromosome) of components. These variable components are analogous to Genes. Thus a chromosome (individual) is composed of several genes (variable components).



Fitness Score

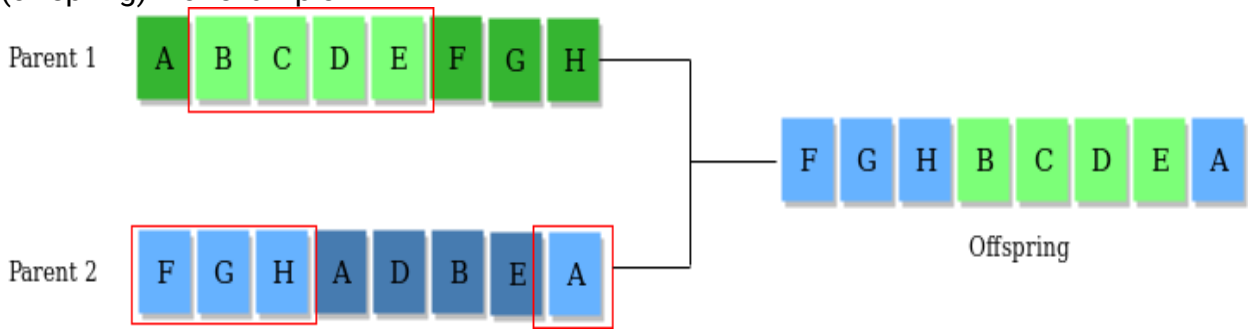
A Fitness Score is given to each individual which **shows the ability of an individual to "compete"**. The individual having optimal fitness score (or near optimal) are sought.
The GAs maintains the population of n individuals (chromosome/solutions) along with their fitness scores.The individuals having better fitness scores are given more chance to reproduce than others. The individuals with better fitness scores are selected who mate and produce **better offspring** by combining chromosomes of parents. The population size is static so the room has to be created for new arrivals. So, some individuals die and get replaced by new arrivals eventually creating new generation when all the mating opportunity of the old population is exhausted. It is hoped that over successive generations better solutions will arrive while least fit die.
Each new generation has on average more "better genes" than the individual (solution) of previous generations. Thus each new generations have better **"partial solutions"** than previous generations. Once the offsprings produced having no significant difference than offspring produced by previous populations, the population is converged. The algorithm is said to be converged to a set of solutions for the problem.

Operators of Genetic Algorithms

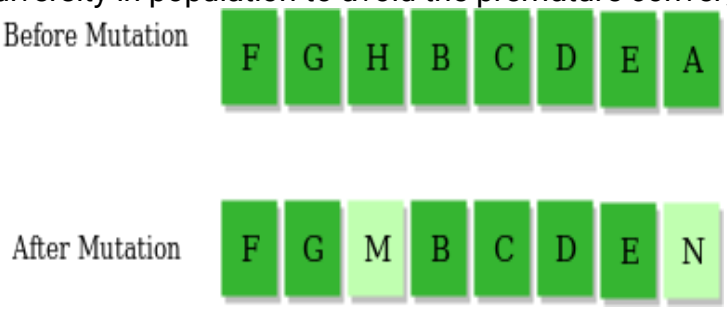
Once the initial generation is created, the algorithm evolve the generation using following operators –

1) Selection Operator: The idea is to give preference to the individuals with good fitness scores and allow them to pass there genes to the successive generations.

2) Crossover Operator: This represents mating between individuals. Two individuals are selected using selection operator and crossover sites are chosen randomly. Then the genes at these crossover sites are exchanged thus creating a completely new individual (offspring). For example –



3) Mutation Operator: The key idea is to insert random genes in offspring to maintain the diversity in population to avoid the premature convergence. For example –



The whole algorithm can be summarized as –

```
1) Randomly initialize populations p

2) Determine fitness of population

3) Untill convergence repeat:

      a) Select parents from population

      b) Crossover and generate new population

      c) Perform mutation on new population

      d) Calculate fitness for new population
```

Neural Networks

Neural networks are artificial systems that were inspired by biological neural networks. These systems learn to perform tasks by being exposed to various datasets and examples without any task-specific rules. The idea is that the system generates identifying characteristics from the data they have been passed without being programmed with a pre-programmed understanding of these datasets.
Neural networks are based on computational models for threshold logic. Threshold logic is a combination of algorithms and mathematics. Neural networks are based either on the study of the brain or on the application of neural networks to artificial intelligence. The work has led to improvements in finite automata theory.

Components of a typical neural network involve neurons, connections, weights, biases, propagation function, and a learning rule. Neurons will receive an input from predecessor neurons that have an activation , threshold , an activation function f, and an output function . Connections consist of connections, weights and biases which rules how neuron transfers output to neuron . Propagation computes the input and outputs the output and sums the predecessor neurons function with the weight. The learning rule modifies the weights and thresholds of the variables in the network.

Supervised vs Unsupervised Learning:
Neural networks learn via supervised learning; Supervised machine learning involves an input variable *x* and output variable *y*. The algorithm learns from a training dataset. With each correct answers, algorithms iteratively make predictions on the data. The learning stops when the algorithm reaches an acceptable level of performance.
Unsupervised machine learning has input data X and no corresponding output variables. The goal is to model the underlying structure of the data for understanding more about the data. The keywords for supervised machine learning are classification and regression. For unsupervised machine learning, the keywords are clustering and association.
Evolution of Neural Networks:
Hebbian learning deals with neural plasticity. Hebbian learning is unsupervised and deals with long term potentiation. Hebbian learning deals with pattern recognition and exclusive-or circuits; deals with if-then rules.
Back propagation solved the exclusive-or issue that Hebbian learning could not handle. This also allowed for multi-layer networks to be feasible and efficient. If an error was found, the error was solved at each layer by modifying the weights at each node. This led to the development of support vector machines, linear classifiers, and max-pooling. The vanishing gradient problem affects feedforward networks that use back propagation and recurrent neural network. This is known as deep-learning.

Hardware-based designs are used for biophysical simulation and neurotrophic computing. They have large scale component analysis and convolution creates new class of neural computing with analog. This also solved back-propagation for many-layered feedforward neural networks.

Convolutional networks are used for alternating between convolutional layers and max-pooling layers with connected layers (fully or sparsely connected) with a final classification layer. The learning is done without unsupervised pre-training. Each filter is equivalent to a weights vector that has to be trained. The shift variance has to be guaranteed to dealing with small and large neural networks. This is being resolved in Development Networks.

Types of Neural Networks

There are *seven* types of neural networks that can be used.

- The first is a multilayer perceptron which has three or more layers and uses a nonlinear activation function.
- The second is the convolutional neural network that uses a variation of the multilayer perceptrons.
- The third is the recursive neural network that uses weights to make structured predictions.
- The fourth is a recurrent neural network that makes connections between the neurons in a directed cycle. The long short-term memory neural network uses the recurrent neural network architecture and does not use activation function.
- The final two are sequence to sequence modules which uses two recurrent networks and shallow neural networks which produces a vector space from an amount of text. These neural networks are applications of the basic neural network demonstrated below.

For the example, the neural network will work with three vectors: a vector of attributes X, a vector of classes Y, and a vector of weights W. The code will use 100 iterations to fit the attributes to the classes. The predictions are generated, weighed, and then outputted after iterating through the vector of weights W. The neural network handles back propagation.

Tools for Data Mining

Rapid Miner

This is very popular since it is a ready made, open source, no-coding required software, which gives advanced analytics. Written in Java, it incorporates multifaceted data mining functions such as data pre-processing, visualization, predictive analysis, and can be easily integrated with WEKA and R-tool to directly give models from scripts written in the former two. Besides the standard data mining features like data cleansing, filtering, clustering, etc, the software also features built-in templates, repeatable work flows, a professional visualisation environment, and seamless integration with languages like Python and R into work flows that aid in rapid prototyping.

Weka

Weka is a collection of machine learning algorithms for data mining tasks. The algorithms can either be applied directly to a dataset or called from your own Java code. Weka contains tools for data pre-processing, classification, regression, clustering, association rules, and visualization. It is also well-suited for developing new machine learning schemes.

Orange

Python users playing around with data sciences might be familiar with Orange. It is a Python library that powers Python scripts with its rich compilation of mining and machine learning algorithms for data pre-processing, classification, modelling, regression, clustering and other miscellaneous functions. Orange also comes with a

visual programming environment and its workbench consists of tools for importing data, and dragging and dropping widgets and links to connect different widgets for completing the workflow.

R

R is a free software environment for statistical computing and graphics written in C++. R Studio is IDE specially designed for R language.It is one of the leading tools used to do data mining tasks and comes with huge community support as well as packaged with hundreds of libraries built specifically for data mining.

Knime

Primarily used for data preprocessing — i.e. data extraction, transformation and loading, Knime is a powerful tool with GUI that shows the network of data nodes. Popular amongst financial data analysts, it has modular data pipe lining, leveraging machine learning, and data mining concepts liberally for building business intelligence reports.

Rattle

Rattle, expanded to 'R Analytical Tool To Learn Easily', has been developed using the R statistical programming language. The software can run on Linux, Mac OS and Windows, and features statistics, clustering, modelling and visualisation with the computing power of R. Rattle is currently being used in business, commercial enterprises and for teaching purposes in Australian and American universities.

Tanagra

TANAGRA is a free open source data mining software for academic and research purposes. It proposes several data mining methods from exploratory data analysis, statistical learning, machine learning and databases area. TANAGRA is more powerful, it contains some supervised learning but also other paradigms such as clustering, factorial analysis, parametric and non parametric statistics, association rule, feature selection and construction algorithms.The main purpose of Tanagra project is to give researchers and students an easy-to-use **data mining software**, conforming to the present norms of the software development in this domain (especially in the design of its GUI and the way to use it), and allowing to analyse either real or synthetic data.

XL Miner

XLMiner is the only comprehensive data mining add-in for Excel, with neural nets, classification and regression trees, logistic regression, linear regression, Bayes classifier, K-nearest neighbors, discriminant analysis, association rules, clustering, principal components, and more.

XLMiner provides everything you need to sample data from many sources — PowerPivot, Microsoft/IBM/Oracle databases, or spreadsheets; explore and visualize your data with multiple linked charts; preprocess and 'clean' your data, fit data mining models, and evaluate your models' predictive power.

The drawback of XL Miner is that is paid add in for excel but there is 15 day free trial option. The software has great features and its integration in excel makes life easier.