ISSN: 0975-766X
CODEN: IJPTFI
Available Online through ***Review Article***
www.ijptonline.com

PRINCIPLES OF DATABASE ORGANIZATION

Safoora Ansari[1], Shireen Begum[2]
[1]Student, Deccan school of pharmacy (affiliated to OU),
[2]Assistant professor, Deccan school of pharmacy,
Department of pharmaceutics, Deccan school of pharmacy, Dar-us-salam, Aghapura,
Hyderabad-500001, Telangana, India.
Email: saf.ans4@gmail.com

Received on 10-01-2019 *Accepted on: 02-02-2019*

Abstract:

A database is a collection of related data organized in a way that data can be easily accessed, managed and updated. Any piece of information can be data. Database management system is a software that allows creating, defining and manipulating of the database. Different types of database management system exists, with some of them designed for the oversight and proper control of databases that are configured for specific purposes. Here are some examples of the various types of DBMS technology that are currently in use, and some of the basic elements that are part of DBMS software applications. It is often referred by its acronym, DBMS. It is actually a tool used to perform a kind of operation on data in a database. Examples of database management system are Oracle, SQL server, Microsoft access, etc.

Key Words: Database, Database management system, Oracle, SQL, Microsoft access

Introduction

1. Database Concept

Database: A database is defined as an organized collection of data. More specifically, a database is an electronic system that allows data to be easily accessed, managed, and updated.[1]

Database Management Systems (DBMS)

DBMS is the software system that helps in creating and managing the data in the database system. This systems were called **Database management systems (DBMS).** DBMS is the system that allows inserting, updating,

deleting and processing of data. Some of the DBMS developed by software houses are Oracle, Ingress, Sybase, Dbase 3+, Foxbase, Foxpro, MS Access, Dataease, Data flex, [2]

Types of Database Management Systems

There are four structural types of database management systems:

- Hierarchical databases.
- Network databases.
- Relational databases.
- Object-oriented databases

Benefits of DBMS

1. The amount of data redundancy in stored data can be reduced.
2. No more data inconsistencies.
3. Stored data can be shared by single or multiple users.
4. Standards can be set and followed.
5. Data integrity can be maintained. Data integrity refers to the problems of ensuring that database contain only one accurate data.
6. Security of data can be simply implemented.
7. Data independence can be achieved, **i.e.** Data and programs that manipulate the data are two different entities.[3]

Table 1: DBMS V/S RDBMS [4]

DBMS	RDBMS
In DBMS relationship between two tables or files are maintained programmatically	In RDBMS relationship between two tables or files can be specified at the time of table creation
DBMS does not support clients/ server architecture	Most of the RDBMS supports clients/ server architecture
DBMS does not support distributed databases	Most of the RDBMS supports distributed databases
In DBMS there is no security of data	In RDBMS there are multiple levels of security 1. Logging in at O/S level

	2. Command level 3. Object level
Each table is given an extension in DBMS	Many tables are grouped in one database in RDBMS
DBMS may satisfy less than 7 to 8 rules of Dr. E F Codd	RDBMS usually satisfy more than 7 to 8 rules of Dr. E F Codd

Relational Database: Relational database can be thought of as comprehensive tables of data. A Relational database is a collection of tables, each of which is assigned a unique name. Each table consist of a set of attributes and stores of records or tuples. Each record (tuple) in a relational table represents an object identified by a unique key and described by a set of attribute values. An entity relationship model is usually used to model the relational database (fig. 7.2)

Table 2: Representation of relational database.

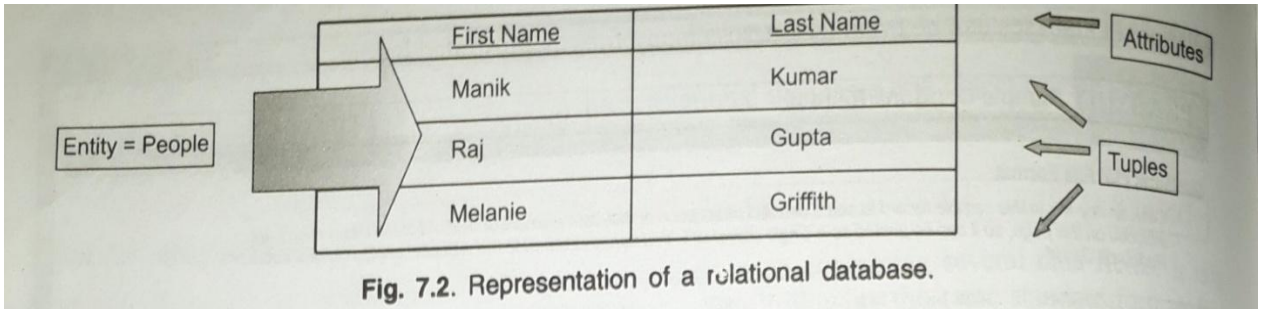


Fig. 7.2. Representation of a relational database.

Relational database is characterized by the following properties:

- Tables represents relations-operations are performed on the tables
- No two tuples can be identical in the database
- Each attribute for a tuple has only one value
- Tuples within a table are unordered

Each tuple is uniquely identified by a primary key.[5]

Database Languages

A DBMS has appropriate languages and interfaces to express database queries and updates.

Database languages can be used to read, store and update the data in the database

Types of database languages:

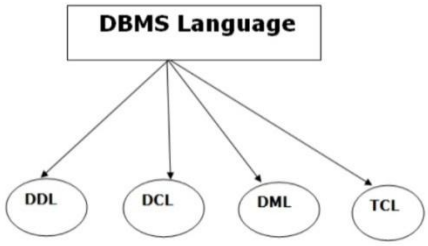


Figure 1.

a) Data Definition Language

b) Data control language

c) Data manipulation language

- Procedural DML
- Non-procedural DML

d) Transaction control language[6]

2. CODD'S RULE

Dr Edgar F. Codd, after his extensive research on the Relational Model of database systems, came up with twelve rules of his own, which according to him, a database must obey in order to be regarded as a true relational database. These rules can be applied on any database system that manages stored data using only its relational capabilities. This is a foundation rule, which acts as a base for all the other rules.

Rule 1: Information Rule

The data stored in a database, may it be user data or metadata, must be a value of some table cell. Everything in a database must be stored in a table format.

Rule 2: Guaranteed Access Rule

Every single data element (value) is guaranteed to be accessible logically with a combination of table-name, primary-key (row value), and attribute-name (column value). No other means, such as pointers, can be used to access data.

Rule 3: Systematic Treatment of NULL Values

The NULL values in a database must be given a systematic and uniform treatment. This is a very important rule because a NULL can be interpreted as one the following − data is missing, data is not known, or data is not applicable.

Rule 4: Active Online Catalogue

The structure description of the entire database must be stored in an online catalog, known as **data dictionary**, which can be accessed by authorized users. Users can use the same query language to access the catalog which they use to access the database itself.

Rule 5: Comprehensive Data Sub-Language Rule

A database can only be accessed using a language having linear syntax that supports data definition, data manipulation, and transaction management operations. This language can be used directly or by means of some application. If the database allows access to data without any help of this language, then it is considered as a violation.

Rule 6: View Updating Rule

All the views of a database, which can theoretically be updated, must also be updatable by the system.

Rule 7: High-Level Insert, Update, and Delete Rule

A database must support high-level insertion, Updation, and deletion. This must not be limited to a single row, that is, it must also support union, intersection and minus operations to yield sets of data records.

Rule 8: Physical Data Independence

The data stored in a database must be independent of the applications that access the database. Any change in the physical structure of a database must not have any impact on how the data is being accessed by external applications.

Rule 9: Logical Data Independence

The logical data in a database must be independent of its user's view (application). Any change in logical data must not affect the applications using it. For example, if two tables are merged or one is split into two different

tables, there should be no impact or change on the user application. This is one of the most difficult rule to apply.

Rule 10: Integrity Independence

A database must be independent of the application that uses it. All its integrity constraints can be independently modified without the need of any change in the application. This rule makes a database independent of the front-end application and its interface.

Rule 11: Distribution Independence: The end-user must not be able to see that the data is distributed over various locations. Users should always get the impression that the data is located at one site only. This rule has been regarded as the foundation of distributed database systems.

Rule 12: Non-Subversion Rule

If a system has an interface that provides access to low-level records, then the interface must not be able to subvert the system and bypass security and integrity constraints.[7]

3. Structure of Database:

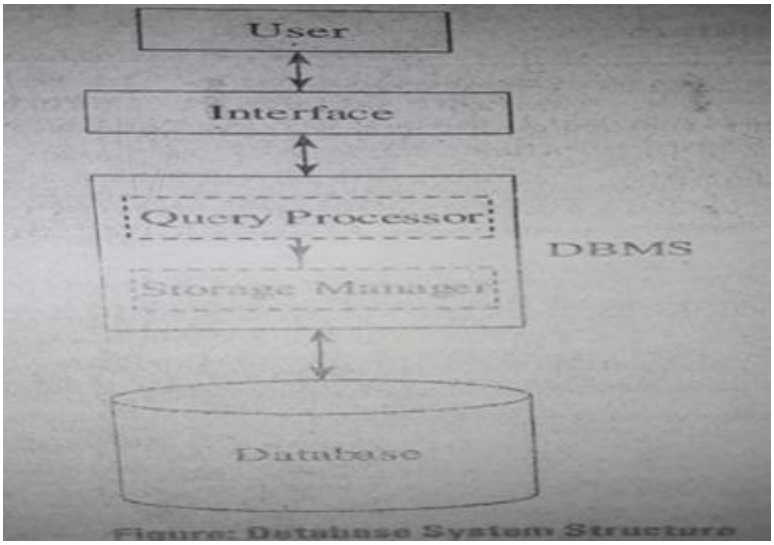


Figure 2: Database system structure.

It means how the data is organized in a structured way. It contains users, query processor, storage manager and database. Users interact with the DBMS through an interface. The DBMS does the processing and retrieves the data from database.

1) Database users:

There are four types of users of database

(A). Database administrator

(B). Designers/ application programmers

(C). End users/ naïve users/ Web users/ agents/ sellers

(D). System analyst

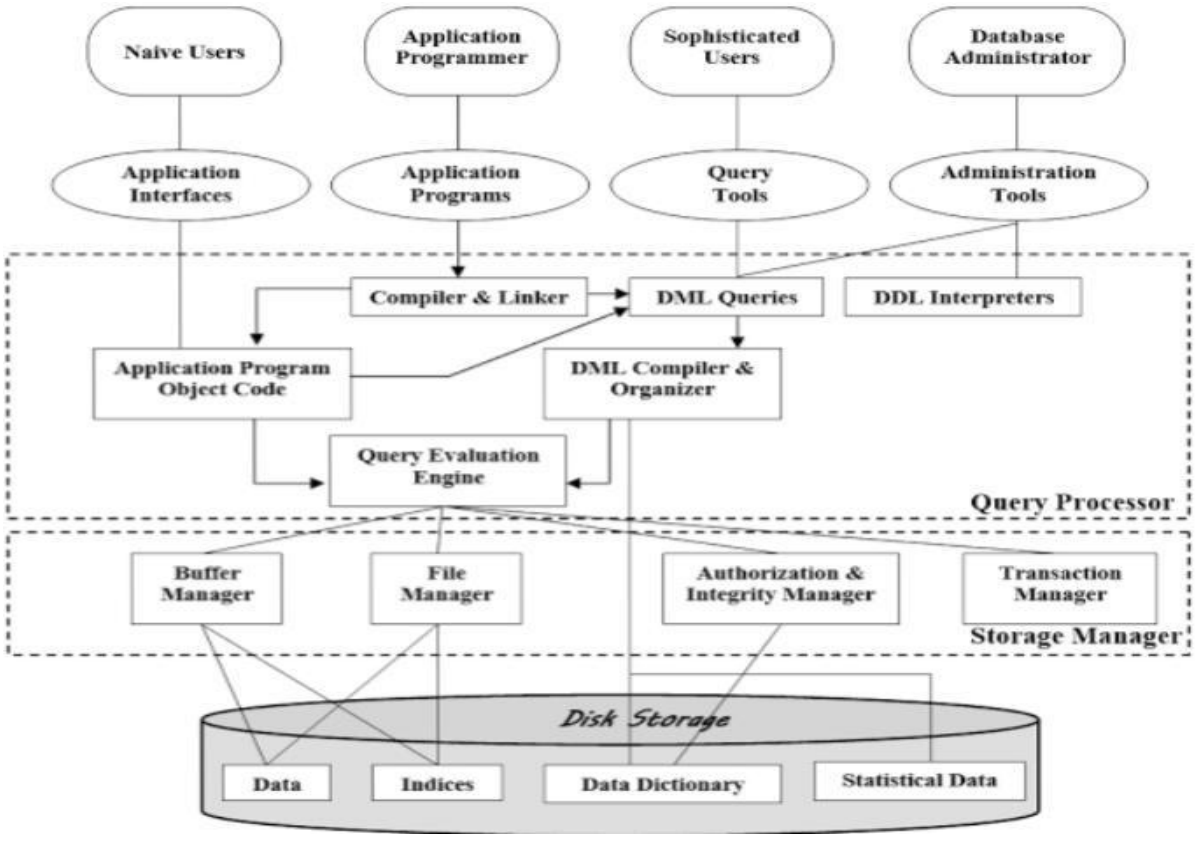
2) Application programming interface

It is a set of procedures that accepts the DML statements and retrieves the result after processing. For **example,** ODBC (Open Database Connectivity) provides an interface for application programs written in C. JDBC(Java Database Connectivity) provides an interface for application program written in JAVA.

3) Database system is divided into two modules.

- Query processing
- Storage Management. [8]

Figure-3: Database structure.



Database Architecture

Database architecture is the set of specifications, rules, and processes that dictate how data is stored in a database and how data is accessed by components of a system. It includes data types, relationships, and naming conventions. The database Architecture describes the organization of all database objects and how they work together. It affects integrity, reliability, scalability and performance.

It is logically divided in two types:

a) Logical two-tier clients / server architecture.

b) Logical three-tier clients / server architecture.

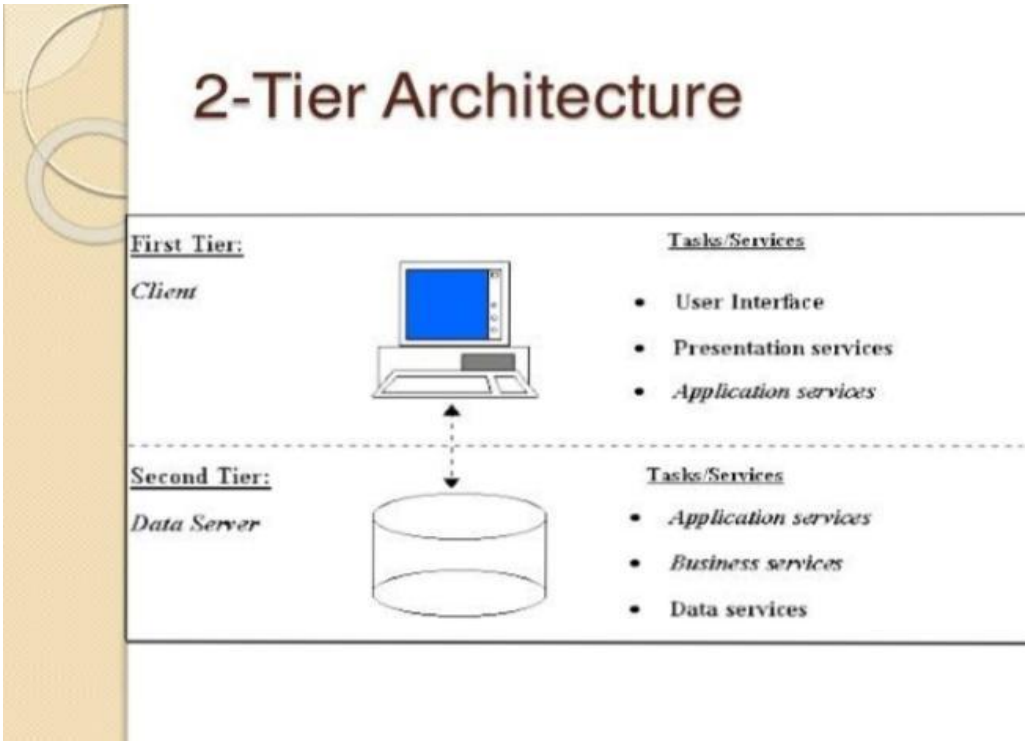


Figure-4.

Two- tier client.

Two-tier client architecture is used for user interface program and application program that runs on client side. An interface called ODBC (open database connectivity) provides an API that allows client side program to call the DBMS. Most DBMS vendors provides ODBC drivers. A client program may connect to several DBMS's. In this architecture some variation of client is also possible for example in some DBMS's more functionality is transferred to the client including data dictionary, optimization etc. Such clients are called data server.

Three-tier client / server architecture

It is commonly used architecture for Web applications. Intermediate layer called application server or Web server stores the Web connectivity software and the business logic part of application used to access the right amount of data from the database sever. This layer act like medium for sending partially processed data between the database server and the client. The database Architecture involves anything that defines the nature of data, the structure of the data, or how the data flows. It consist of three levels: external, conceptual, and internal.

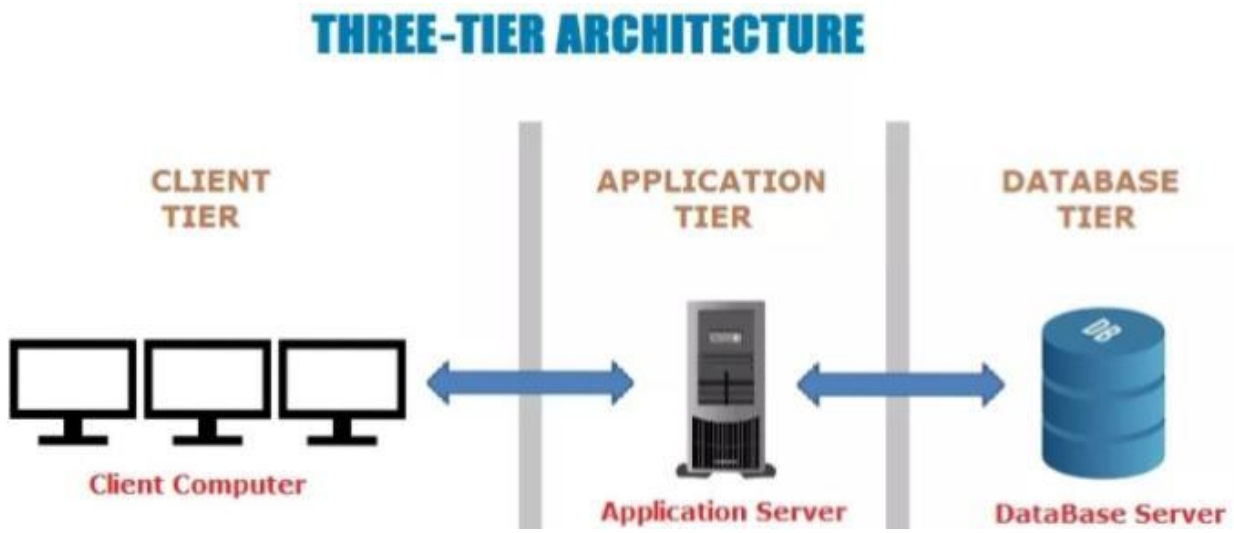


Figure 5. Three-tire architecture.

External level (highest)

External level is the users view of the database. It consist of a numbers of different eternal views of the database. It also describes art of the database for particular group of users. This highest level provides a powerful and flexible security mechanism by hiding parts of database from certain users. The user is not aware of the existence of any attributes that are missing from the view. It permits users to access data in a way that is customized to their needs, so that the same data can be seen by different users in a different way, at the same time.

Conceptual level

The logical structure of the entire database administrator. It includes about what data is stored in the database, the relationships among the data and complete view of the data requirements of the organization, independent of any storage considerations. It represents

- Entries, attribute relations.
- Constrains on data
- Semantic information on data
- Security, integrity information.
- Supports each external view: any data available to a user must be contained in, or derivable from the conceptual level.

Internal level

This level includes physical representation of the database on the computer and how the data is stored in the database.

Physical implementation of the database to achieve optimal run-time performance and storage space utilization involves:

- Storage space allocation For data and indexes
- Record description for storage
- Record placement
- Data compression and Encryption [9]

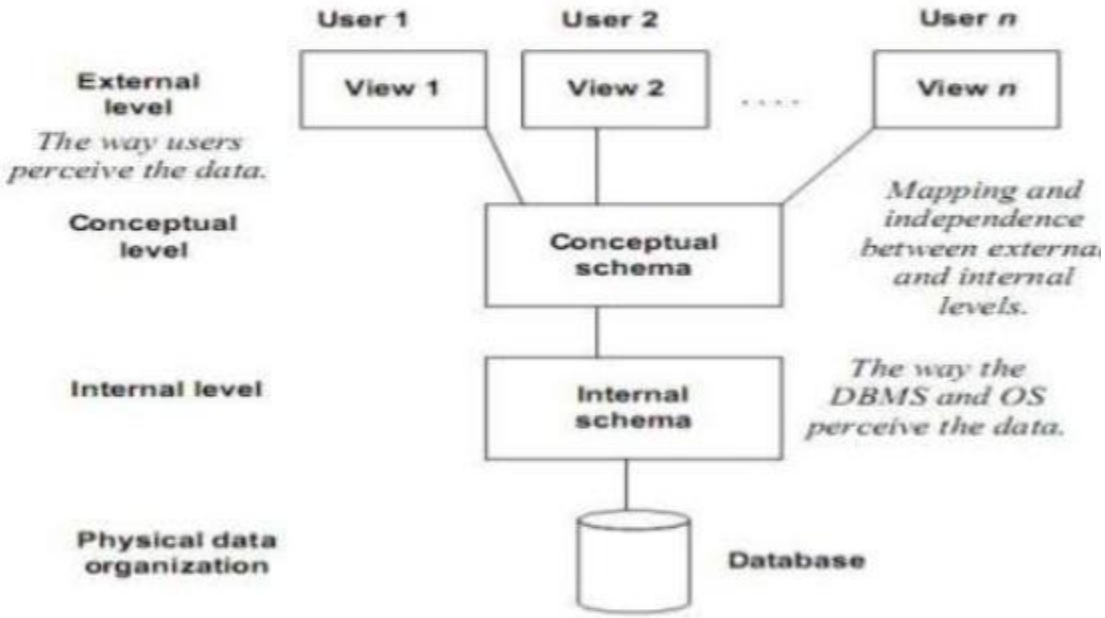


Figure 6: Three level architecture.

5. Normalization

Is a process of organizing the data in database to avoid data redundancy, insertion anomaly, update anomaly & deletion anomaly. Let's discuss about anomalies first then we will discuss normal forms with examples.

Anomalies in DBMS

There are three types of anomalies that occur when the database is not normalized. These are – Insertion, update and deletion anomaly.

Update anomaly: If we have two rows in a table for employee Rick as he belongs to two departments of the company. If we want to update the address of Rick then we have to update the same in two rows or the data will become inconsistent. If somehow, the correct address gets updated in one department but not in other then as per the database, Rick would be having two different addresses, which is not correct and would lead to inconsistent data.

Insert anomaly: Suppose a new employee joins the company, who is under training and currently not assigned to any department then we would not be able to insert the data into the table if emp_dept field doesn't allow nulls.

Delete anomaly: Suppose, if at a point of time the company closes the department D890 then deleting the rows that are having emp_dept as D890 would also delete the information of employee Maggie since she is assigned only to this department. To overcome these anomalies we need to normalize the data. In the next section we will discuss about normalization.

Normalization

Here are the most commonly used normal forms:

- First normal form(1NF)
- Second normal form(2NF)
- Third normal form(3NF)
- Boyce & Codd normal form (BCNF)
- Forth normal form(4NF)
- Fifth normal form(5NF)

First Normal Form

First Normal Form is defined in the definition of relations (tables) itself. This rule defines that all the attributes in a relation must have atomic domains. The values in an atomic domain are indivisible units. **Table 3 (1NF).**

Course	Content
Programming	Java, c++
Web	HTML, PHP, ASP

We re-arrange the relation (table) as below, to convert it to First Normal Form.

Table 4 (1NF rearrangement)

Course	Content
Programming	Java
Programming	c++
Web	HTML
Web	PHP
Web	ASP

Each attribute must contain only a single value from its pre-defined domain.

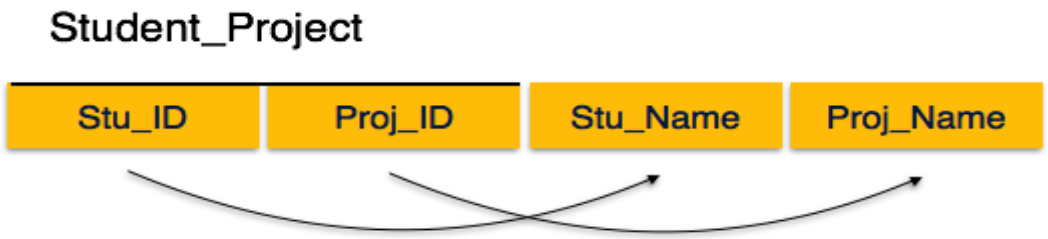
Second Normal Form

Before we learn about the second normal form, we need to understand the following −

- **Prime attribute** − An attribute, which is a part of the candidate-key, is known as a prime attribute.
- **Non-prime attribute** − An attribute, which is not a part of the prime-key, is said to be a non-prime attribute.

If we follow second normal form, then every non-prime attribute should be fully functionally dependent on prime key attribute. That is, if X → A holds, then there should not be any proper subset Y of X, for which Y → A also holds true.

Table 5 (2NF).



We see here in Student_Project relation that the prime key attributes are Stu_ID and Proj_ID. According to the rule, non-key attributes, i.e. Stu_Name and Proj_Name must be dependent upon both and not on any of the prime key attribute individually. But we find that Stu_Name can be identified by Stu_ID and Proj_Name can be identified by Proj_ID independently. This is called partial dependency, which is not allowed in Second Normal Form.

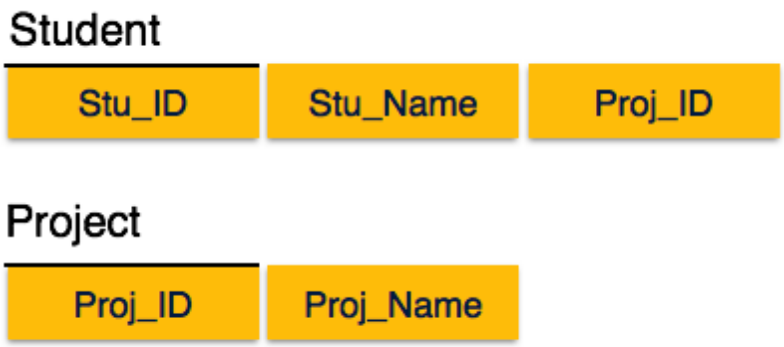


Table 6 (rearrangement of 2NF).

We broke the relation in two as depicted in the above picture. So there exists no partial dependency.

Third Normal Form

For a relation to be in Third Normal Form, it must be in Second Normal form and the following must satisfy No non-prime attribute is transitively dependent on prime key attribute.

For any non-trivial functional dependency, X → A, then either −

a. X is a superkey or,

b. A is prime attribute.

Table 7 (3NF).

We find that in the above Student_detail relation, Stu_ID is the key and only prime key attribute. We find that City can be identified by Stu_ID as well as Zip itself. Neither Zip is a superkey nor is City a prime attribute. Additionally, Stu_ID → Zip → City, so there exists transitive dependency.

To bring this relation into third normal form, we break the relation into two relations as follows −

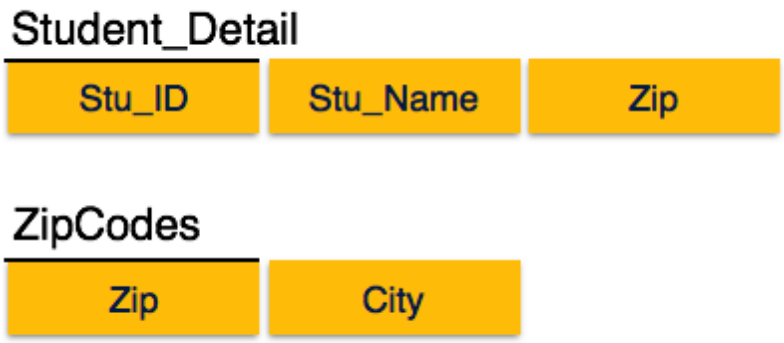


Table 8: Rearrangement of 3NF.

Boyce-Codd Normal Form

Boyce-Codd Normal Form (BCNF) is an extension of Third Normal Form on strict terms. BCNF states that

For any non-trivial functional dependency, X → A, X must be a super-key.

In the above image, Stu_ID is the super-key in the relation Student_Detail and Zip is the super-key in the relation ZipCodes. So,

Stu_ID → Stu_Name, Zip

and

Zip → City

Which confirms that both the relations are in BCNF.

4NF (Fourth Normal Form) Rules

If no database table instance contains two or more, independent and multivalued data describing the relevant entity, then it is in 4th Normal Form.

5NF (Fifth Normal Form) Rules

A table is in 5th Normal Form only if it is in 4NF and it cannot be decomposed into any number of smaller tables without loss of data.

6NF (Sixth Normal Form) Proposed

6th Normal Form is not standardized, yet however, it is being discussed by database experts for some time. Hopefully, we would have a clear & standardized definition for 6th Normal Form in the near future...[10]]

6. Data Mining and Knowledge Discovery in Databases: Data mining is an important step in an overall knowledge discovery process. Knowledge Discovery process involves selection and sampling of the appropriate data from the database; preprocessing and cleaning of the data to remove redundancies, errors, and conflicts; Transforming and reducing data to a format more suitable for the data mining; evaluation of the mined data; and visualization of the evaluation results.

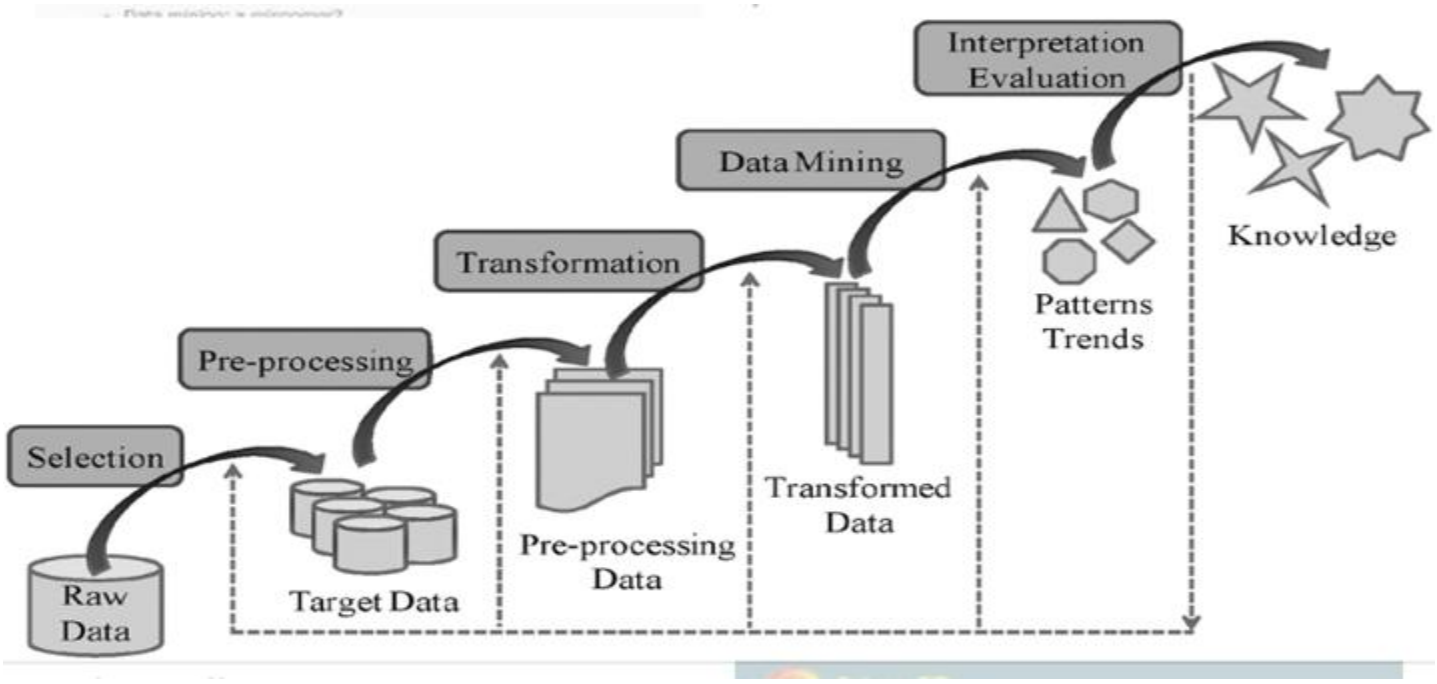


Figure 7: Data mining.

Usually many interactions of the knowledge discovery process are required. Similarly, each phase of the knowledge discovery process has associated challenges, as outline here.

Preprocessing and cleaning

Major part of the work is in preparing the data for the actual analysis associated with the data mining. The major preparatory activities include the following activities:

i. **Data characterization** – creating a high level description of the nature and the contents of the data to be mined.
ii. **Consistency analysis –** determining the statistical variability in the data, independent of the domain.
iii. **Domain analysis—**validating the data values in the large context of the biology.
iv. **Data enrichment –** drawing from multiple data sources to minimizes the limitations of a single data source.
v. **Frequency and distribution analysis -** weighing values as a function of their frequency of occurrence
vi. **Normalization -** Transforming data values from one representation to another.
vii. **Missing value analysis -** detecting, characterizing, and dealing with missing data values.

Transforming and reduction

In the transformation and reduction phase, data sets are reduced to minimum size possible through sampling or summary statistics. For example, tables of data may be replaced by descriptive statistics such as mean standard deviation.

Data mining method

The process of data mining is concerned with extracting patterns from the data by using various techniques. Some of these techniques are defined below.

a. Classification methods involves mapping data into one of several predefined or newly discovered classes.
b. Regression method involves assigning data a continuous numerical variable based on statistical methods
c. Link analysis methods involves evaluating apparent connections or links between data in the database

d. Deviation detection is used to identify data values that are outside of the norms, as defined by existing model or by evaluating the ordering of observation.

e. Segmentation techniques identifies classes or group of data that behaves similarly, according to an established metric.

Evaluation

In the evaluation phase of knowledge discovery, the patterns identified by the data mining analysis are interpreted. Evaluation ranges from simple statistical analysis and complex numerical analysis of sequences and structures to determining the biological implications of the findings.

Visualization

Visualization of evaluation results can range from simple pie chart to 3-D virtual reality displays.[11]

7. Bibliographic Database:

A bibliographic database is a database of bibliographic records, an organized digital collection of references to publish literature, including journal and newspaper articles, conference proceedings, reports, government and legal Publications, patents, books etc. In contrast to library catalogue entries, a large proportion of the bibliographic records in bibliographic databases describe articles, conference papers, etc. rather than complete monographs, and they generally contain very which subject descriptions in the form of keywords, subject classification terms, or abstract. [12]

8. Drug Information Database:

Drug information database means providing clinically relevant information through; reading ; study, or practical experience on chemical substance that is used in diagnosis prevention and treatment of a disease. It covers all types of information including; objective and subjective information as well as information gathered by scientific observation or practical experience.

Some drug information database:

- Health square
- Mayo clinic
- Drug info.net

- Drug.com
- DailyMed
- Medline plus
- Pharma web, etc.
- Pubchem
- Rx med
- Medscape
- Medicine net
- WebMd
- Drug bank
- Consumer oriented drug base
- EMC [13]

CONCLUSION:

- A DBMS is a systematic operational system consist of tools making data, data saving, and data manipulating an easier task. DBMS is majorly used throughout the world for data handling.
- The goal of DBMS is to offer more convenience as well as more efficiency to access data from a database with high security.
- DBMS is perhaps most useful for providing a centralized view from multiple users, from multiple locations in a controlled manner.
- A DBMS can limit what data the end user sees, as well as how that end users can view the data, providing many views of single database schema.

Acknowledgements:

The successful accomplishment of this seminar would not have been possible but by the timely help and guidance rendered by many people. I would like to mention few of them. My first salutation goes to Almighty Allah and my Parents for being ever so kind and courteous. It gives me an immense pleasure to acknowledge a

debt of gratitude to my guide Ms. SHIREEN BEGUM Dept of Pharmaceutics, Deccan School of Pharmacy for her constant encouragement, suggestions, supervision and support. I would like to express profound gratitude to Syed Abdul Azeez Basha, honourable Principal of Deccan School of Pharmacy, Hyderabad, for guiding us as well as providing us the support to conduct this seminar.

References:

1. [1], [2], [3], [4] Text book of SQL, PL/SQL the programming language of ORACLE (4th edition) by "IVAN BAYROSS." BPB Publications.
2. [5], [11]Bioinformatics concepts, skills and applications (2nd edition) by S. C. RASTOGI, Namita Mendiratta, Parag Rastogi. CBS publishers and distributors pvt. Ltd.
3. https://www.google.co.in/url?sa=t&source=web&rct=j&url=https://www.w3schools.in/dbms/database-languages/&ved=2ahUKEwjYou-A_-DfAhUHLo8KHTm0B_wQFjACegQIDhAO&usg=AOvVaw3vJrQTDceu26qwIcz8rk42&cshid=1547047214144.
4. https://www.google.co.in/url?sa=t&source=web&rct=j&url=https://www.webopedia.com/amp/TERM/C/Codds_Rules.html&ved=2ahUKEwikzqnx_-DfAhXKK48KHSfkDvsQFjAGegQIChAe&usg=AOvVaw1GuteyIyzHuynHTwgwlNLD&ampcf=1
5. https://www.google.co.in/url?sa=t&source=web&rct=j&url=https://www.tutorialcup.com/dbms/structure-of-dbms.htm&ved=2ahUKEwjY2LuZgOHfAhWM6Y8KHSXsBW0QFjARegQIBBAB&usg=AOvVaw2tiHiI7C1Wya16M8BZIPNF&cshid=1547047619670.
6. https://www.google.co.in/url?sa=t&source=web&rct=j&url=https://www.studytonight.com/dbms/architecture-of-database.php&ved=2ahUKEwitjIeCgeHfAhVMto8KHcJhCM4QFjAUegQIAxAB&usg=AOvVaw1D0XtCXGkHo-sQLBkrJt3j&cshid=1547047889340.
7. https://www.google.co.in/url?sa=t&source=web&rct=j&url=https://www.tutorialspoint.com/dbms/database_normalization.htm&ved=2ahUKEwjdzI-

iguHfAhWMqI8KHQIVCmsQFjAAegQIAhAB&usg=AOvVaw1RcexNm-

hVxhkyubUwv7V1&cshid=1547048079085.

8. http://www.newworldencyclopedia.org/entry/Bibliographic_database

9. https://www.google.co.in/url?sa=t&source=web&rct=j&url=https://www.clinicaladvisor.com/drug-information-database/section/3727/&ved=2ahUKEwiBgqqMg-HfAhUE448KHXjOBGsQFjABegQIAhAB&usg=AOvVaw2e4QcjgG7eYxRZEtIBhjeN&cshid=1547048330434.

Corresponding Author:

Safoora Ansari[*],

Student, Deccan school of pharmacy (affiliated to OU),

Email: saf.ans4@gmail.com