Unit-4: Database Management System

Contents: Introduction to DBMS, DBMS Models, SQL, Database Design and Data Security, Data Warehouse, Data Mining, Database Administrator

What is Database?

Database is a collection of related data and data is a collection of facts and figures that can be processed to produce information.

Mostly data represents recordable facts. Data aids in producing information, which is based on facts. For example, if we have data about marks obtained by all students, we can then conclude about toppers and average marks.

A database management system stores data in such a way that it becomes easier to retrieve, manipulate, and produce information.

Characteristics of DBMS

Real-world entity: A modern DBMS is more realistic and uses real-world entities to design its architecture. It uses the behavior and attributes too. For example, a school database may use students as an entity and their age as an attribute.

Relation-based tables: DBMS allows entities and relations among them to form tables. A user can understand the architecture of a database just by looking at the table names.

Isolation of data and application: A database system is entirely different than its data. A database is an active entity, whereas data is said to be passive, on which the database works and organizes. DBMS also stores metadata, which is data about data, to ease its own process.

Less redundancy: DBMS follows the rules of normalization, which splits a relation when any of its attributes is having redundancy in values. Normalization is a mathematically rich and scientific process that reduces data redundancy.

Consistency: Consistency is a state where every relation in a database remains consistent. There exist methods and techniques, which can detect attempt of leaving database in inconsistent state. A DBMS can provide greater consistency as compared to earlier forms of data storing applications like file-processing systems.

Query Language: DBMS is equipped with query language, which makes it more efficient to retrieve and manipulate data. A user can apply as many and as different filtering options as required to retrieve a set of data. Traditionally it was not possible where file-processing system was used

ACID Properties: DBMS follows the concepts of Atomicity, Consistency, Isolation, and Durability (normally shortened as ACID). These concepts are applied on transactions, which manipulate data in a database. ACID properties help the database stay healthy in multi-transactional environments and in case of failure.

Multiuser and Concurrent Access: DBMS supports multi-user environment and allows them to access and manipulate data in parallel. Though there are restrictions on transactions when users attempt to handle the same data item, but users are always unaware of them.

Multiple views: DBMS offers multiple views for different users. A user who is in the Sales department will have a different view of database than a person working in the Production department. This feature enables the users to have a concentrate view of the database according to their requirements.

Security: Features like multiple views offer security to some extent where users are unable to access data of other users and departments. DBMS offers methods to impose constraints while entering data into the database and retrieving the same at a later stage. DBMS offers many different levels of security features, which enables multiple users to have different views with different features.

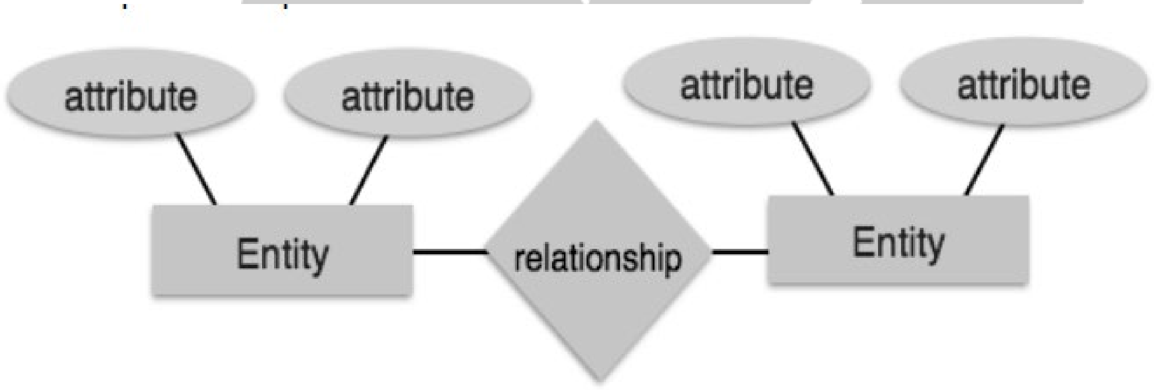
Database Models

Data models define how the logical structure of a database is modeled. Data Models are fundamental entities to introduce abstraction in a DBMS. Data models define how data is connected to each other and how they are processed and stored inside the system.

Two types

Entity-Relationship Model

Entity-Relationship (ER) Model is based on the notion of real-world entities and relationships among them. While formulating real-world scenario into the database model, the ER Model creates entity set, relationship set, general attributes, and constraints.



[*Image: ER Model*]

ER Model is best used for the conceptual design of a database.

ER Model is based on:

- Entities and their attributes.

- An entity in an ER Model is a real-world entity having properties called attributes. Every attribute is defined by its set of values called domain.

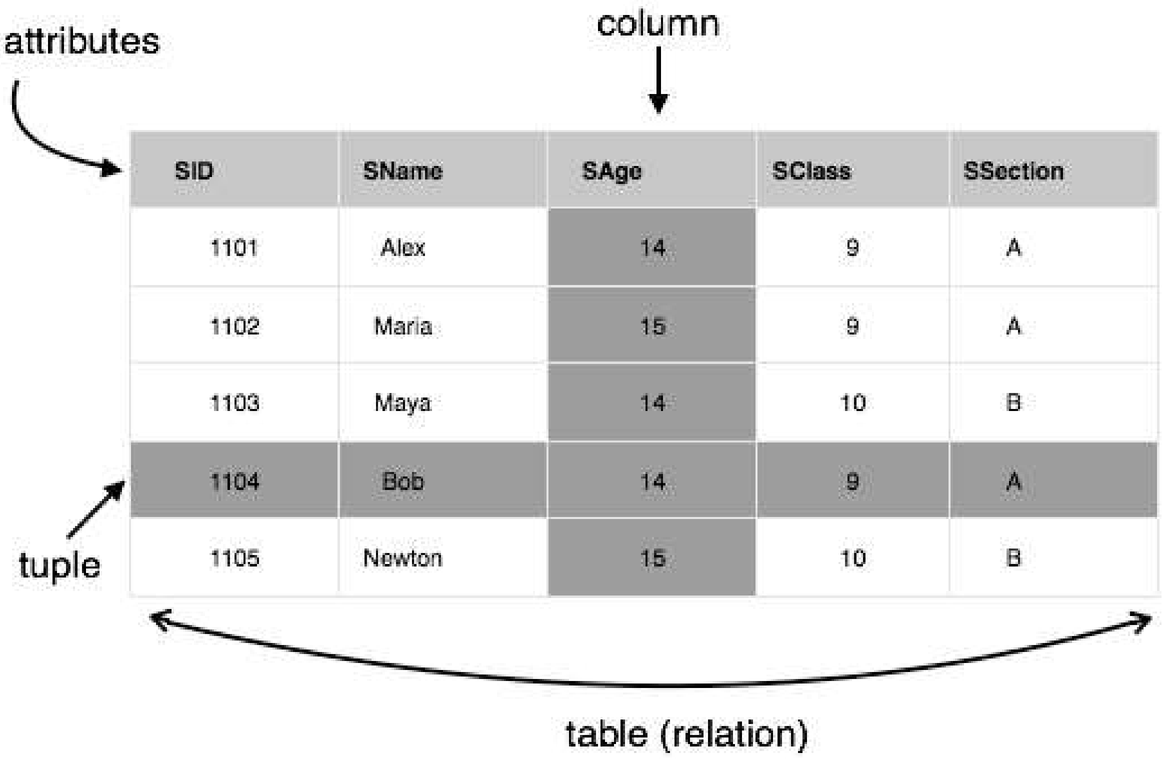
For example, in a school database, a student is considered as an entity. Student has various attributes like name, age, class, etc.

Relationships among entities

The logical association among entities is called relationship. Relationships are mapped with entities in various ways. Mapping cardinalities define the number of associations between two entities.

Relational Model

The most popular data model in DBMS is the Relational Model. It is more scientific model than others. This model is based on first-order predicate logic and defines a table as an n-ary relation.



SID	SName	SAge	SClass	SSection
1101	Alex	14	9	A
1102	Maria	15	9	A
1103	Maya	14	10	B
1104	Bob	14	9	A
1105	Newton	15	10	B

The main highlights of this model are:

1. Data is stored in tables called relations.
2. Relations can be normalized.
3. In normalized relations, values saved are atomic values.
4. Each row in a relation contains a unique value
5. Each column in a relation contains values from a same domain.

6.

SQL (Structured Query Language)

What is SQL?

SQL is Structured Query Language, which is a computer language for storing, manipulating and retrieving data stored in relational database. SQL is the standard language for Relation Database System. All relational database management systems like MySQL, MS Access, Oracle, Sybase, Informix, postgres and SQL Server use SQL as standard database language.

Why SQL?

Allows users to access data in relational database management systems.

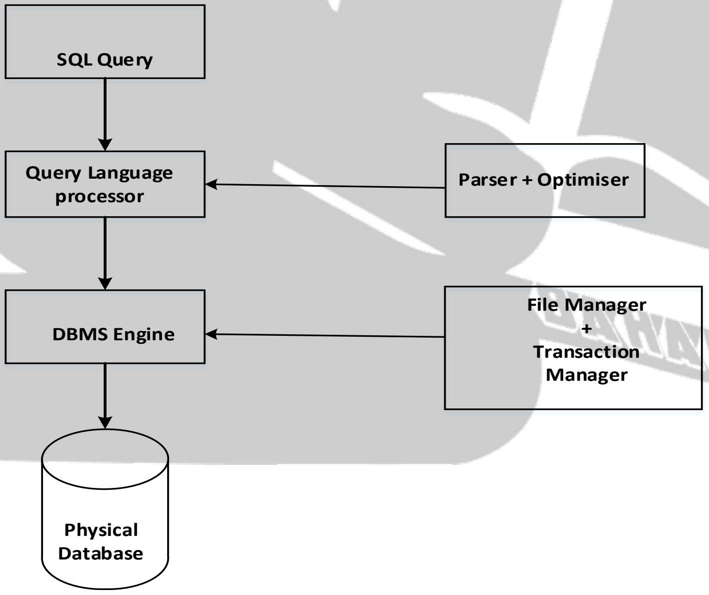
Allows users to describe the data.

Allows users to define the data in database and manipulate that data.

Allows to embed within other languages using SQL modules, libraries & pre-compilers.

Allows users to create and drop databases and tables

SQL Process



When you are executing an SQL command for any RDBMS, the system determines the best way to carry out your request and SQL engine figures out how to interpret the task.

There are various components included in the process. These components are Query Dispatcher, Optimization Engines, Classic Query Engine and SQL Query Engine, etc. Classic query engine handles all non-SQL queries, but SQL query engine won't handle logical files

SQL Commands

The standard SQL commands to interact with relational databases are CREATE, SELECT, INSERT, UPDATE, DELETE and DROP. These commands can be classified into groups based on their nature.

DDL	Data Definition Language:
CREATE	Creates a new table, a view of a table, or other object in database
ALTER	Modifies an existing database object, such as a table.
DROP	Deletes an entire table, a view of a table or other object in the database.
DML	Data Manipulation Language:
INSERT	Creates a record
UPDATE	Modifies records
DELETE	Deletes records
DCL	Data Control Language:
GRANT	Gives a privilege to user
REVOKE	Takes back privileges granted from user
DQL	Data Query Language:
SELECT	Retrieves certain records from one or more tables

Database concepts

A database is a collection of logically related records.

A relational database stores its data in 2-dimensional tables.

A table is a two-dimensional structure made up of rows (tuples, records) and columns (attributes, fields).

Example: a table of students engaged in sports activities, where a student is allowed to participate in at most one activity

Student ID	Activity	Fee

100	Skiing	200
150	Swimming	50
175	Squash	50
200	Swimming	50

Table Characteristics

1. Field

Every table is broken up into smaller entities called fields. The fields in the CUSTOMERS table consist of Student ID, Activity, Fee.

A field is a column in a table that is designed to maintain specific information about every record in the table.

2. Record (Row)

A record, also called a row of data, is each individual entry that exists in a table. For example, there are 4 records in the above table. Following is a single row of data or record in the CUSTOMERS table.

3. Column

each column has a unique attribute name

each column (attribute) description (metadata) is stored in the database

order is unimportant

all entries in a column have the same data type

4. Primary Keys

a primary key is an attribute or a collection of attributes whose value(s) uniquely identify each row in a relation

Student ID	Activity	Fee
100	Skiing	200
150	Swimming	50
175	Squash	50
200	Swimming	50

a primary key should be minimal: it should not contain unnecessary attributes

we assume that a student is allowed to participate in at most one activity then the only possible primary key in the above table is StudentID

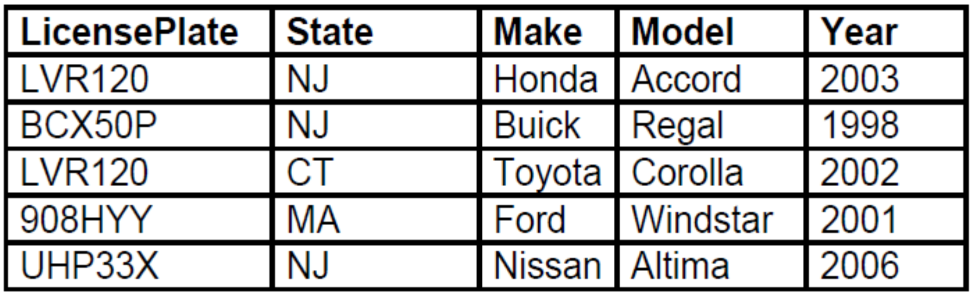
Sometimes there is more than one possible choice. Each possible choice is called a candidate key

If we allow the students to participate in more than one activity then the only possible primary key is the combined value of StudentID and Activity

such a multi-attribute primary key is called a composite key or concatenated key

5. Composite Keys

A table can only have one primary key but sometimes the primary key can be made up of several fields



LicensePlate	State	Make	Model	Year
LVR120	NJ	Honda	Accord	2003
BCX50P	NJ	Buick	Regal	1998
LVR120	CT	Toyota	Corolla	2002
908HYY	MA	Ford	Windstar	2001
UHP33X	NJ	Nissan	Altima	2006

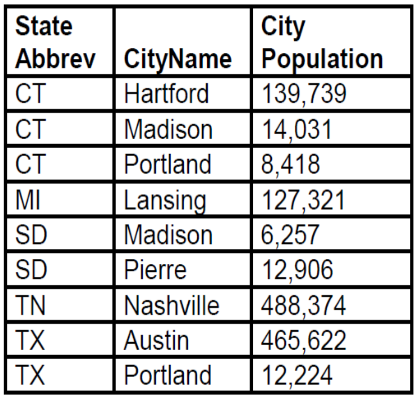
Concatenation means putting two things next to one another.

License Plate is not a possible primary key, because two different cars can have the same license plate number if they're from different states but if we concatenate LicensePlate and State, the resulting value of (LicensePlate, State) must be unique.

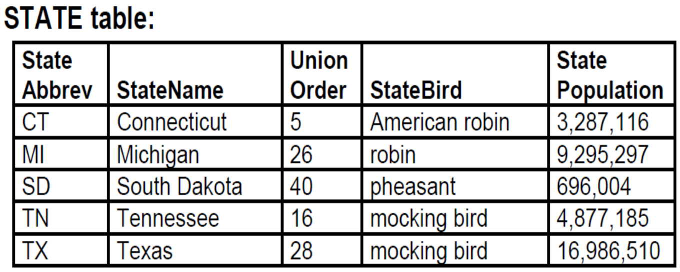
example: “LVR120NJ” and “LVR120CT”

6. Foreign Key

A foreign key is an attribute or a collection of attributes whose value are intended to match the primary key of some related record (usually in a different table)



State Abbrev	CityName	City Population
CT	Hartford	139,739
CT	Madison	14,031
CT	Portland	8,418
MI	Lansing	127,321
SD	Madison	6,257
SD	Pierre	12,906
TN	Nashville	488,374
TX	Austin	465,622
TX	Portland	12,224



STATE table:

State Abbrev	StateName	Union Order	StateBird	State Population
CT	Connecticut	5	American robin	3,287,116
MI	Michigan	26	robin	9,295,297
SD	South Dakota	40	pheasant	696,004
TN	Tennessee	16	mocking bird	4,877,185
TX	Texas	28	mocking bird	16,986,510

primary key in STATE table: StateAbbrev

primary key in CITY table: (StateAbbrev, CityName)

foreign key in CITY relation: StateAbbrev

Database Design

Database design provides a means to represent real world entities in a form that can be processed by the computer. Database models present a process of abstracting real world entities into computer representations.

To develop a good design, one has to understand the meaning of information and the intended use of stored representation within the computer system. Once we develop the understanding and identify the use of information in the application, we can determine how much and what kind of information we require.

After determination of application’s information requirement, it will be clear that which data entities represent information redundancies, entities that are critical, useful and are not related to the applications.

It is important to collect and analyze the static and dynamic information available about real world application before starting the database design.

For evolving a good database design, it is important that one uses a model, a database design model. The database design models have following benefits.

They provide a means to represent real-world objects in computer usable form

They capture and represent associations and relationships among the real-world objects, allowing the application designers to capture the dynamic nature of the real-world enterprise’s activities.

They define how the objects in the application interact in logical terms

They allow the database designer to capture static and dynamic organization and flow of information within the modeled enterprise.

They allow designers and users of the system to better understand the static and dynamic behavior of the system being modeled.

They help in improving the maintainability, scalability and reliability of the system.

Steps of Database Design

1. Requirement analysis

   To determine how to construct the DBMS for an application, the designer must first determine the scope of the problem requiring the database system.

   Requirement analysis are used to define the scope of the requirement of an application

   It includes

   Defining the human factors of the application

   Defining the application's functionality

   Defining all the information managed and used by the application

   Determining from where to where all interfaces to an application are derived

   Identifying all the resource requirements including hardware, software and other physical resources.

   Deciding on the security requirements and mechanisms

   Defining the quality, reliability, performance and operational aspect of the application.

2. Information Modeling

   The objective of information modeling is to identify the major entities that are fundamental in an application and model them in the target database schema model

   The information collected during the requirement analysis stage forms the input for information modeling. This information will enable the database designer to fully and correctly define the major entities to be modeled in the database

   The attributes that define the entities of the application are grouped together according to the data model used and stored for further reference.

3. Design Constraints

   The database systems require certain controls and limits for it to truly represent the real-world system behavior.

   These limits or controls are called constraints in database parlance

   There are many kinds of database constraints as follows

   a. Structural Constraint
   b. Type Constraint
   c. Range Constraint

d. Relationship Constraint
e. Temporal Constraint

1. Structural Constraint

   The structure of the information within the database gives an idea about entities in the database.

   For example, simple data structures are represented using simple structures while complex data structures will need advanced structures.

   Structural constraints are specified to force the placement of information into structures that best matches the application

2. Type constraints

   A type constraint limits the application to only one representation of information for an entity’s attribute.

   For example, the database designer might want to limit the name attribute to a fixed length character string, the age attribute to a number etc. Type constraints allow a limitation of the range of information representations that an attribute can have.

3. Range Constraints

   Range constraints can limit the values an attribute can take. It refers to the possible values that a particular data item can have. Range constraints can be used to limit the value of a particular attribute within a range.

   For example, We can specify that the employee numbers should be in the range 1000-9999.

4. Relational constraints

   These constraints represent relationships on values between entities. For example, there could be a relationship constraint between the entities Manager and Employee that the maximum bonus of manager should not be greater than six times that of the employee

5. Temporal Constraints

   These constraints indicate the time period for which some information is valid. For example, the value of attribute sale tax or exercise duty is valid for a specific period. Once the period is over, new values will come into effect.

Database Security

Database security involves protecting a database from unauthorized access, malicious destruction and even any accidental loss or misuse. Due to the high value of data incorporate databases, there is strong motivation for unauthorized users to gain access to it, for instance, competitors or dissatisfied employees

The competitors may have strong motivation to access confidential information about product development plans, cost-saving initiatives and customer profiles.

Some may want to access information regarding unannounced financial results, business transactions and even customers credit card numbers. They may not only steal the valuable information, in fact, if they have access to the database, they may even destroy it and great havoc may occur.

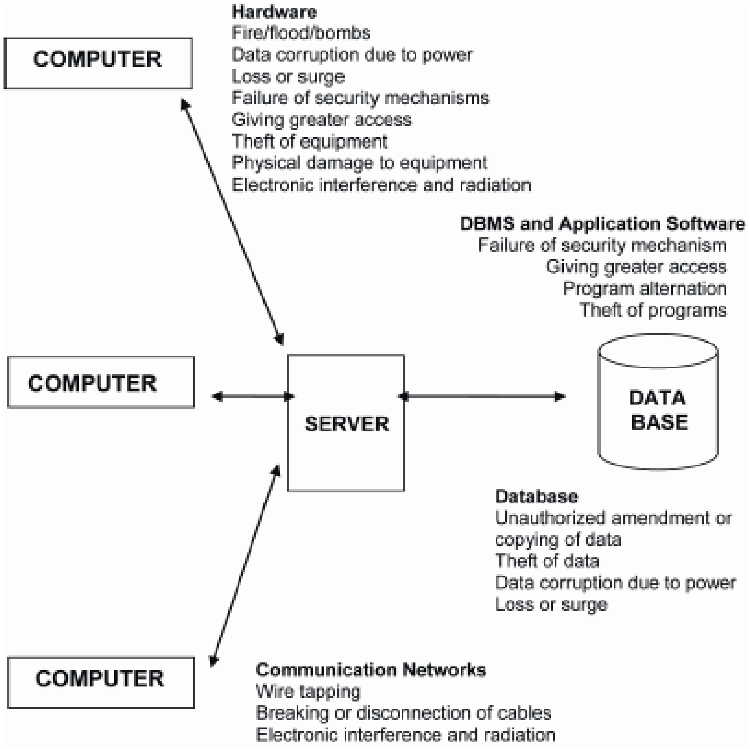

Hardware
Fire/flood/bombs
Data corruption due to power
Loss or surge
Failure of security mechanisms
Giving greater access
Theft of equipment
Physical damage to equipment
Electronic interference and radiation
COMPUTER
DBMS and Application Software
Failure of security mechanism
Giving greater access
Program alternation
Theft of programs
COMPUTER
SERVER
DATA BASE
Database
Unauthorized amendment or
copying of data
Theft of data
Data corruption due to power
Loss or surge
COMPUTER
Communication Networks
Wire tapping
Breaking or disconnection of cables
Electronic interference and radiation

Fig: Threats to Computer System

There are various ways how we can secure our system. The types of computer-based controls to threats on computer systems range from physical controls to administrative policies and procedures.

1. Authorization

   Authorization is the granting of a right or privilege that enables a subject to have legitimate access to a system or a system's object.

Usually, a user or subject can gain access to or a system through individual user accounts where each user is given a unique identifier, which is used by the operating system to determine that they have the authorization to do so.

2. Access Control

   Access controls to a database system is based on the granting and revoking of privileges. A privilege allows a user to create or access (that is read, write or modify) a database object or to execute a DBMS utility.

   The DBMS keeps track of how these privileges are granted to users and possibly revoked, and ensures that at all times only users with necessary privileges can access an object.

3. Views

   A view is created by querying one or more of the base tables, producing a dynamic result table for the user at the time of the request. The user may be allowed to access the view but not the base tables which the view is based. The view mechanism hides some parts of the database from certain users and the user is not aware of the existence of any attributes or rows that are missing from the view.

4. Backup and recovery

   Backup is the process of periodically taking a copy of the database and log file to offline storage media. Backup is very important for a DBMS to recover the database following a failure or damage.

5. Encryption

   Encryption is the process of encoding of the data using a special algorithm that renders the data unreadable by any program without the decryption key . Data encryption can be used to protect highly sensitive data like customer credit card numbers or user password. Some DBMS products include encryption routines that would automatically encode the sensitive data when they are stored or transmitted over communication channels

6. RAID (Redundant Array of Independent Disks)

   The DBMS should continue to operate even though if one of the hardware components fails. The hardware that the DBMS is running on must be fault-tolerant where the DBMS should continue operating and processing even if there is hardware failure.

The main hardware components that should be fault-tolerant are disk drives, disk controllers, CPU, power supplies and cooling fans

Data Warehouse

Data Warehouse is a collection of data designed to support management decision making. The primary goal of a data warehouse is providing access to the data of an organization, data consistency, capacity to separate and combine data, inclusion of tools set to query, analyze and present information, publishing user data, driving business engineering etc.

A data warehouse essentially combines information from several sources into one comprehensive database. For example, in the business world, a data warehouse might incorporate customer information from a company's point-of-sale systems (the cash registers), its website, its mailing lists and its comment cards. Alternatively, it might incorporate all the information about employees, including time cards, demographic data, salary information, etc.

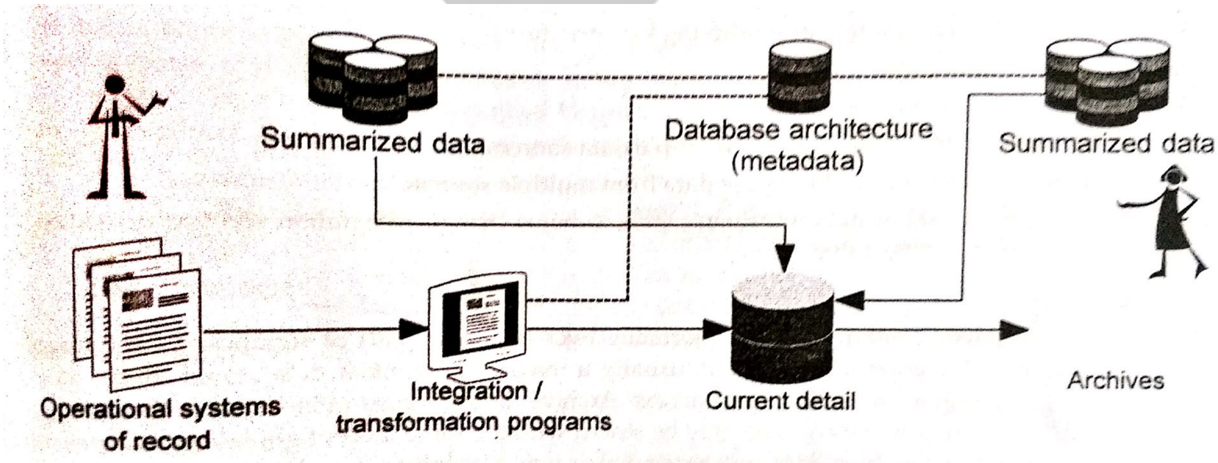


Fig: Concept of Data Warehouse

By combining all of this information in one place, a company can analyze its customers in a more holistic way, ensuring that it has considered all the information available. Data warehousing also makes data mining possible, which is the task of looking for patterns in the data that could lead to higher sales and profits.

The collection of data used by data warehouse may be characterized as subject-oriented, integrated, non-volatile and time-variants.

1. Subject Oriented

   Data is arranged and optimized to provide answer to questions from diverse functional areas. Data is organized and summarized by topic like Sales/Marketing/Finance/Distribution etc.

2. Integrated

The data warehouse is a centralized, consolidated database that integrates data derived from the entire organization.

3. Multiple Sources
4. Diverse Sources
5. Diverse Formats
6. Time Variant

   The Data Warehouse represents the flow of data through time. It contains projected data from statistical models. Data is periodically uploaded then time-dependent data is recomputed.

7. Non-Volatile

   Once data is entered it is NEVER removed. It represents the company's entire history–Near term history is continually added to it. It is always growing and must support terabyte databases and multiprocessors. It is Read-Only database for data analysis and query processing

Data Warehouse Architecture

The main benefits of implementing a data warehouse are cost-effective decision-making, better business intelligence, enhanced customer service, business re-engineering, information system re-engineering etc.

Data Mining

Data mining refers to extracting or mining knowledge from large amounts of data. It is the computational process of discovering patterns in large data sets involving methods at the intersection of artificial intelligence, machine learning, statistics, and database systems.

The overall goal of the data mining process is to extract information from a data set and transform it into an understandable structure for further use.

The key properties of data mining are

1. Automatic discovery of patterns
2. Prediction of likely outcomes
3. Creation of actionable information
4. Focus on large datasets and databases

Given databases of sufficient size and quality, data mining technology can generate new business opportunities by providing following capabilities

1. Automated prediction of trends and behaviors

   Data mining automates the process of finding predictive information in large databases. A typical example of a predictive problem is targeted marketing. Data mining uses data on past promotional mailings to identify the targets most likely to maximize return on investment in future mailings.

2. Automated discovery of previously unknown patterns

   Data mining tools sweep through databases and identify previously hidden patterns in one step. An example of pattern discovery is the analysis of retail sales data to identify seemingly unrelated products that are often purchased together. Other pattern discovery problems include detecting fraudulent credit card transactions and identifying anomalous data that could represent data entry keying errors

Tasks of Data Mining

Data mining involves six common classes of tasks;

1. Anomaly detection (Outlier/change/deviation detection)

   The identification of unusual data records, that might be interesting or data errors that require further investigation.

2. Association rule learning (Dependency modelling)

   It searches for relationships between variables. For example a supermarket might gather data on customer purchasing habits. Using association rule learning, the supermarket can determine which products are frequently bought together and use this information for marketing purposes. This is sometimes referred to as market basket analysis.

3. Clustering

   It is the task of discovering groups and structures in the data that are in some way or another "similar", without using known structures in the data.

4. Classification

   It is the task of generalizing known structure to apply to new data. For example, an e-mail program might attempt to classify an e-mail as "legitimate" or as "spam".

5. Regression

   It attempts to find a function which models the data with the least error.

6. Summarization

   It provides more compact representation of the data set, including visualization and report generation

Architecture of Data Mining

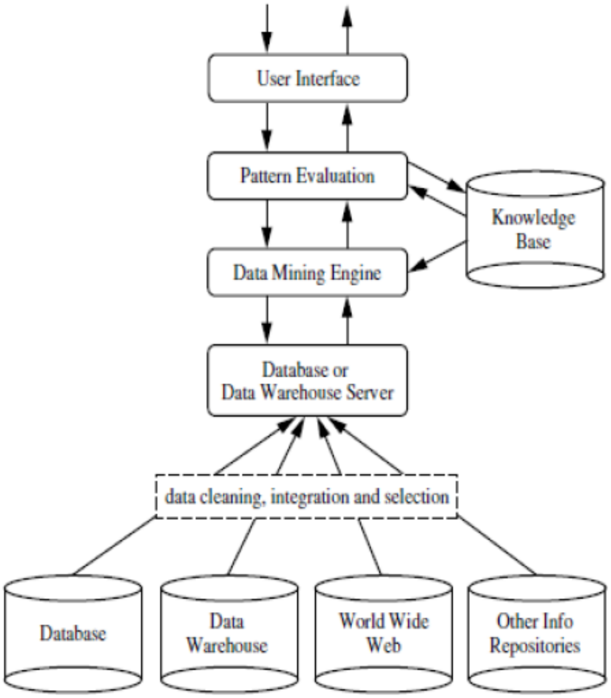


Fig: Architecture of Data Mining

1. Knowledge Base:

   This is the domain knowledge that is used to guide the search or evaluate the interestingness of resulting patterns. Such knowledge can include concept hierarchies, used to organize attributes or attribute values into different levels of abstraction.

2. Data Mining Engine:

   This is essential to the data mining system and ideally consists of a set of functional modules for tasks such as characterization, association and correlation analysis, classification, prediction, cluster analysis, outlier analysis, and evolution analysis

3. Pattern Evaluation Module:

   This component typically employs interestingness measures interacts with the data mining modules so as to focus the search toward interesting patterns. It may use interestingness thresholds to filter out discovered patterns. For efficient data mining, it is highly recommended to push the evaluation of pattern interestingness as deep as possible into the mining process so as to confine the search to only the interesting patterns.

4. User interface:

   This module communicates between users and the data mining system, allowing the user to interact with the system by specifying a data mining query or task, providing information to help focus the search, and performing exploratory data mining based on the intermediate data mining results.

   In addition, this component allows the user to browse database and data warehouse schemas or data structures, evaluate mined patterns, and visualize the patterns in different forms

Database Administrator

A database administrator (DBA) is a specialized computer systems administrator who maintains a successful database environment by directing or performing all related activities to keep the data secure.

The top responsibility of a DBA is to maintain data integrity. This means the DBA will ensure that data is secure from unauthorized access but is available to users.

DBA is responsible for backing up systems in case of power outages or other disasters. A DBA is also frequently involved in tasks related to training employees in database management and use, designing, implementing, and maintaining the database system and establishing policies and procedures related to the organization's data management policy.

Database administrator can be classified as

1. System DBA Overview

   System DBAs typically have a background in system architecture and are responsible for the physical and technical aspects of a database. This can include installing upgrades and patches to fix program bugs and ensuring that the database works properly in a firm's computer system.

2. Application DBA Overview

   Application DBAs use complex programming languages to write or debug programs that work with the database. Usually this database has been designed for a specific application or a set of applications, such as customer service software.