Database Systems Lecture 1

Before the Advent of Database Systems

The way in which computers manage data has come a long way over the last few decades.Today's users take for granted the many benefits found in a database system. However, itwasn't that long ago that computers relied on a much less elegant and costly approach to data management called the file-based system.

File-based System

One way to keep information on a computer is to store it in permanent files. A company system has a number of application programs; each of them is designed to manipulate data files. These application programs have been written at the request of the users in the organization. New applications are added to the system as the need arises. The system just described is called the *file-based system*.

Consider a traditional banking system that uses the file-based system to manage the organization's data shown in Figure 1.1. As we can see, there are different departments in the bank. Each has its own applications that manage and manipulate different data files.For banking systems, the programs may be used to debit or credit an account, find the balance of an account, add a new mortgage loan and generate monthly statements.

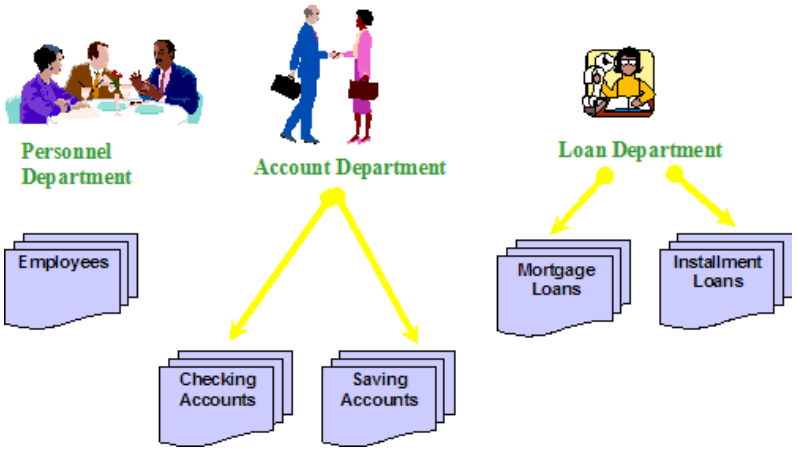


Figure 1.1. Example of a file-based system used by banks to manage data.

Disadvantages of the file-based approach

Using the file-based system to keep organizational information has a number of disadvantages. Listed below are five examples.

Data redundancy

Often, within an organization, files and applications are created by different programmersfrom various departments over long periods of time. This can lead to *data redundancy*, a situation that occurs in a database when afield needs to be updated in more than one table. This practice can lead to several problems such as:

- Inconsistency in data format
- The same information being kept in several different places (files)

Data inconsistency, a situation where various copies of the same data are conflicting, wastes storage space and duplicates effort

Data isolation

Data isolation is a property that determines when and how changes made by oneoperation become visible to other concurrent users and systems. This issue occurs in a concurrency situation. This is a problem because:

- It is difficult for new applications to retrieve the appropriate data, which might be stored in various files.

Integrity problems

Problems with *data integrity* is another disadvantage of using a file-based system. Itrefers to the maintenance and assurance that the data in a database are correct andconsistent. Factors to consider when addressing this issue are:

- Data values must satisfy certain consistency constraints that are specified in theapplication programs.
- It is difficult to make changes to the application programs in order to enforce newconstraints.

Security problems

Security can be a problem with a file-based approach because:

- There are constraints regarding accessing privileges.
- Application requirements are added to the system in an ad-hoc manner so it isdifficult to

enforce constraints.

Concurrency access

Concurrency is the ability of the database to allow multiple users access to the same recordwithout adversely affecting transaction processing. A file-based system must manage, or prevent, concurrency by the application programs. Typically, in a file-based system, whenan application opens a file, that file is locked. This means that no one else has access to the file at the same time.

In database systems, concurrency is managed thus allowing multiple users access to the same record. This is an important difference between database and file-based systems.

Database Approach

The difficulties that arise from using the file-based system have prompted the development of a new approach in managing large amounts of organizational informationcalled the *database approach*.

Databases and database technology play an important role in most areas where computers are used, including business, education and medicine. To understand the fundamentals of database systems, we will start by introducing some basic concepts in this area.

The meaning of data

Data are factual information such as measurements or statistics about objects and concepts. We used at for discussions or as part of a calculation. Data can be a person, a place, an event, an action or any one of a number of things. A single fact is an element of data, or a *data element*. If data are information and information is what we are in the business of working with, you can start to see where you might be storing it. Data can be stored in:

- Filing cabinets
- Spreadsheets
- Folders
- Ledgers
- Lists
- Piles of papers on your desk

All of these items store information, and so too does a database. Because of the mechanical nature of databases, they have terrific power to manage and process the information they hold. This can make the information they house much more useful for your work.

With this understanding of data, we can start to see how a tool with the capacity to storea collection of data and organize it, conduct a rapid search, retrieve and process, might make a difference to how we can use

data.

Data versus information

Information and data are not the same thing. Information is sometimes defined as processed data, but simply because data has been processed does not mean that it is useful to an organization. The average price of corn in Iowa may be meaningful to thousands of farmers but useless to an automobile manufacturer. **Data** is defined simply as numbers, words, names, and other symbols that can be stored in a computer system. **Information** is simply useful data. If a single data value (such as the price of a competitor'sproduct) is useful, then it is information. Several data values may be manipulated or processed to create information (such as in computing the average price of several competing products). If the resulting data value is not useful, then it is not information, no matter how much it has been processed.

Fundamentals Concepts:

What Is a Database?

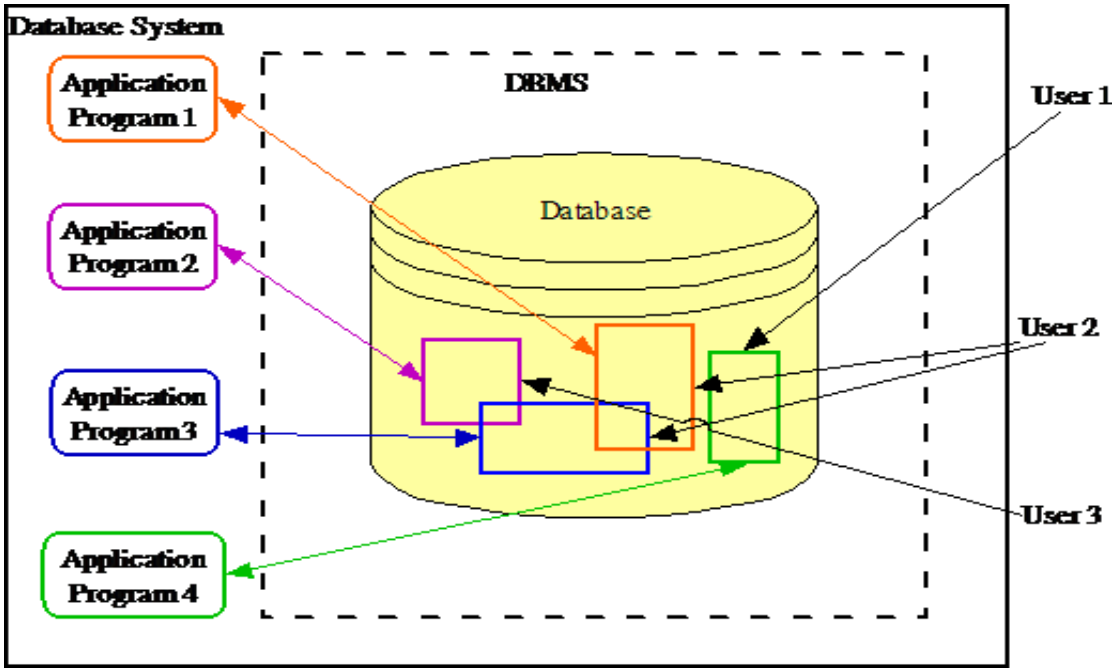


Figure 2.1. A database is a repository of data.

A *database* is a shared collection of related data used to support the activities of a particular organization. A database can be viewed as a repository of data that is defined once and then accessed by various users as shown in Figure 2.1.

Database Properties

A database has the following properties:

- It is a representation of some aspect of the real world or a collection of *dataelement*s (facts) representing real-world information.
- A database is logical, coherent and internally consistent.
- A database is designed, built and populated with data for a specific purpose.
- Each data item is stored in a field.
- A combination of fields makes up a *table*. For example, each field in an employeetable contains data about an individual employee.

Database Management System

A *database management system (DBMS)* is a collection of programs that enables users tocreate and maintain databases and control all access to them. The primary goal of a DBMSis to provide an environment that is both convenient and efficient for users to retrieve and store information.

With the database approach, we can have the traditional banking system as shown in Figure 2.3. In this bank example, a DBMS is used by the Personnel Department, the Account Department and the Loan Department to access the shared corporate database.

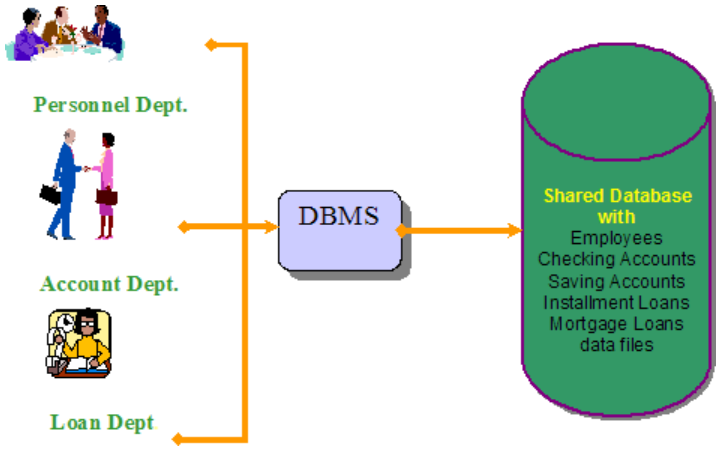


Figure 2.3. A bank database management system (DBMS).

Characteristics and Benefits of a Database

There are a number of characteristics that distinguish the database approach from the file-based system or approach. This section describes the benefits (and features) of the database system.

Self-describing nature of a database system

A database system is referred to as *self-describing* because it not only contains the database itself, but also *metadata* which defines and describes the data and relationshipsbetween tables in the database. This information is used by the DBMS software or database users if needed. This separation of data and information about the data makes a database system totally different from the traditional file-based system in which the data definition is part of the application programs.

Insulation between program and data

In the file-based system, the structure of the data files is defined in the application programs so if a user wants to change the structure of a file, all the programs that accessthat file might need to be changed as well. On the other hand, in the database approach,the data structure is stored in the system catalogue and not in the programs. Therefore, one change is all that is needed to change the structure of a file. This insulation between the programs and data is also called program-data independence.

Support for multiple views of data

A database supports multiple views of data. A *view* is a subset of the database, which isdefined and dedicated for particular users of the system. Multiple users in the system might have different views of the system. Each view might contain only the data ofinterest to a user or group of users. DBMS permits many users to have access to its database either individually or simultaneously. It is not important for users to be aware of how and where the data they access is stored.

Sharing of data and multiuser system

Current database systems are designed for multiple users. That is, they allow many usersto access the same database at the same time. This access is achieved through features called *concurrency control strategies*. These strategies ensure that the data accessed are always correct and that data integrity is maintained.

The design of modern multiuser database systems is a great improvement from those in the past which restricted usage to one person at a time.

Control of data redundancy

In the database approach, ideally, each data item is stored in only one place in the database. In some cases,

data redundancy still exists to improve system performance, but such redundancy is controlled by application programming and kept to minimum by introducing as little redundancy as possible when designing the database.

Data sharing

The integration of all the data, for an organization, within a database system has many advantages. First, it allows for data sharing among employees and others who have accessto the system. Second, it gives users the ability to generate more information from a given amount of data than would be possible without the integration.

Enforcement of integrity constraints

Database management systems must provide the ability to define and enforce certain constraints to ensure that users enter valid information and maintain data integrity. A *database constraint* is a restriction or rule that dictates what can be entered or edited in a table such as a postal code using a certain format or adding a valid city in the City field.

There are many types of database constraints. *Data type*, for example, determines the sort of data permitted in afield, for example numbers only. *Data uniqueness* such as theprimary key ensures that no duplicates are entered. Constraints can be simple (field based) or complex (programming).

Restriction of unauthorized access

Not all users of a database system will have the same accessing privileges. For example, one user might have *read-only access* (i.e., the ability to read a file but not make changes),while another might have *read and write privileges*, which is the ability to both read and modify a file. For this reason, a database management system should provide a security subsystem to create and control different types of user accounts and restrict unauthorized access.

Transaction processing

A database management system must include concurrency control subsystems. This feature ensures that data remains consistent and valid during transaction processing evenif several users update the same information.

Backup and recovery facilities

Backup and recovery are methods that allow you to protect your data from loss. The database system

provides a separate process, from that of a network backup, for backingup and recovering data. If a hard drive fails and the database stored on the hard drive is not accessible, the only way to recover the database is from a backup.

If a computer system fails in them idle of a complex update process, the recoverysubsystem is responsible for making sure that the database is restored to its original state.These are two more benefits of a database management system.

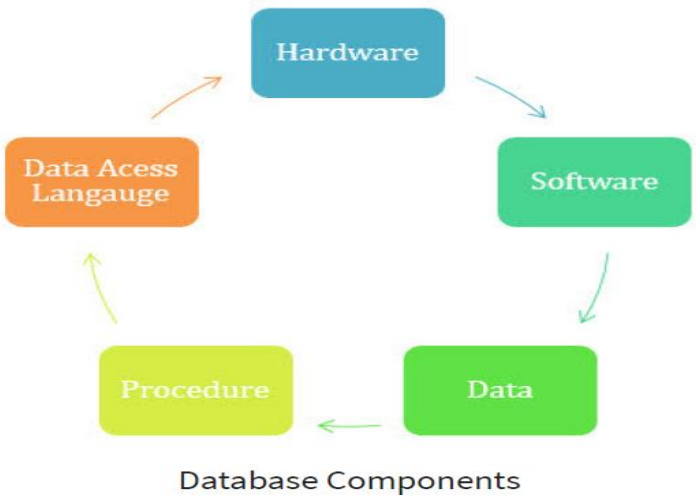
Disadvantage of DBMS

DBMS may offer plenty of advantages but, it has certain flaws:

- Cost of Hardware and Software of a DBMS is quite high which increases the budget of your organization.
- Most database management systems are often complex systems, so the training for users to use the DBMS is required.
- In some organizations, all data is integrated into a single database which can be damaged because of electric failure or database is corrupted on the storage media.
- Use of the same program at a time by many users, sometimes lead to the loss of some data.
- DBMS can't perform sophisticated calculations.

Database Components

There are five main components of a database:



Database Components

Hardware

The hardware consists of physical, electronic devices like computers, I/O devices, storage devices, etc. This offers the interface between computers and real-world systems.

Software

This is a set of programs used to manage and control the overall database. This includes the database software itself, the Operating System, the network software used to share the data among users, and the application programs for accessing data in the database.

Data

Data is a raw and unorganized fact that is required to be processed to make it meaningful. Data can be simple at the same time unorganized unless it is organized. Generally, data comprises facts, observations, perceptions, numbers, characters, symbols, images, etc.

Procedure

Procedure is a set of instructions and rules that help you to use the DBMS. It is designing and running the database using documented methods, which allows you to guide the users who operate and manage it.

Database Access Language

Database Access language is used to access the data to and from the database, enter new data, update already existing data, or retrieve required data from DBMS. The user writes some specific commands in a database access language and submits these to the database.

Data Models

Underlying the structure of a database is the data model: a collection of conceptual tools for describing data, data relationships, data semantics, and consistency constraints.

There are a number of different data models that can be classified into four different categories:

• RelationalModel. The relational model uses a collection of tables to represent both data and the relationships among those data. Each table has multiple columns, and each column has a unique name. Tables are also known as relations. The relational model is an example of a record-based model. Record-based models are so named because the database is structured in fixed-format records of several types. Each table contains records of a particular type. Each record type defines a fixed number of fields, or attributes. The columns of

the table correspond to the attributes of the record type. The relational data model is the most widely used data model, and a vast majority of current database systems are based on the relational model.

• Entity-Relationship Model. The entity-relationship (E-R) data model uses a collection of basic objects, called *entities*, and *relationships* among these objects. An entity is a "thing" or "object" in the real world that is distinguishable from other objects. The entity-relationship model is widely used in database design.

• Semi-structured Data Model. The semi-structured data model permits the specification of data where individual data items of the same type may have different sets of attributes. This is in contrast to the data models mentioned earlier, where every data item of a particular type must have the same set of attributes. *JSON* and *Extensible Markup Language* (*XML*) are widely used semi-structured data representations.

• Object-Based Data Model. Object-oriented programming (especially in Java, C++, or C#) has become the dominant software-development methodology. This led initially to the development of a distinct object-oriented data model, but today the concept of objects is well integrated into relational databases. Standards exist to store objects in relational tables. Database systems allow procedures to be stored in the database system and executed by the database system. This can be seen as extending the relational model with notions of encapsulation, methods, and object identity.