Chapter 1

Introduction to Databases

Chapter 1 - Objectives

- Some common uses of database systems.
- Characteristics of file-based systems.
- Problems with file-based approach.
- Meaning of the term database.
- Meaning of the term Database Management System (DBMS).

Chapter 1 - Objectives

- Typical functions of a DBMS.
- Major components of the DBMS environment.
- Personnel involved in the DBMS environment.
- History of the development of DBMSs.
- Advantages and disadvantages of DBMSs.

Examples of Database Applications

- Purchases from the supermarket
- Purchases using your credit card
- Booking a holiday at the travel agents
- Using the local library
- Taking out insurance
- Renting a video
- Using the Internet
- Studying at university

Manual Filing System

- Physical files labeled and stored in one or more cabinets
- Still used in our homes to store receipts, invoices, warranties, …
- Works well when the amount of item stored is small and no cross-referencing and data processing needed

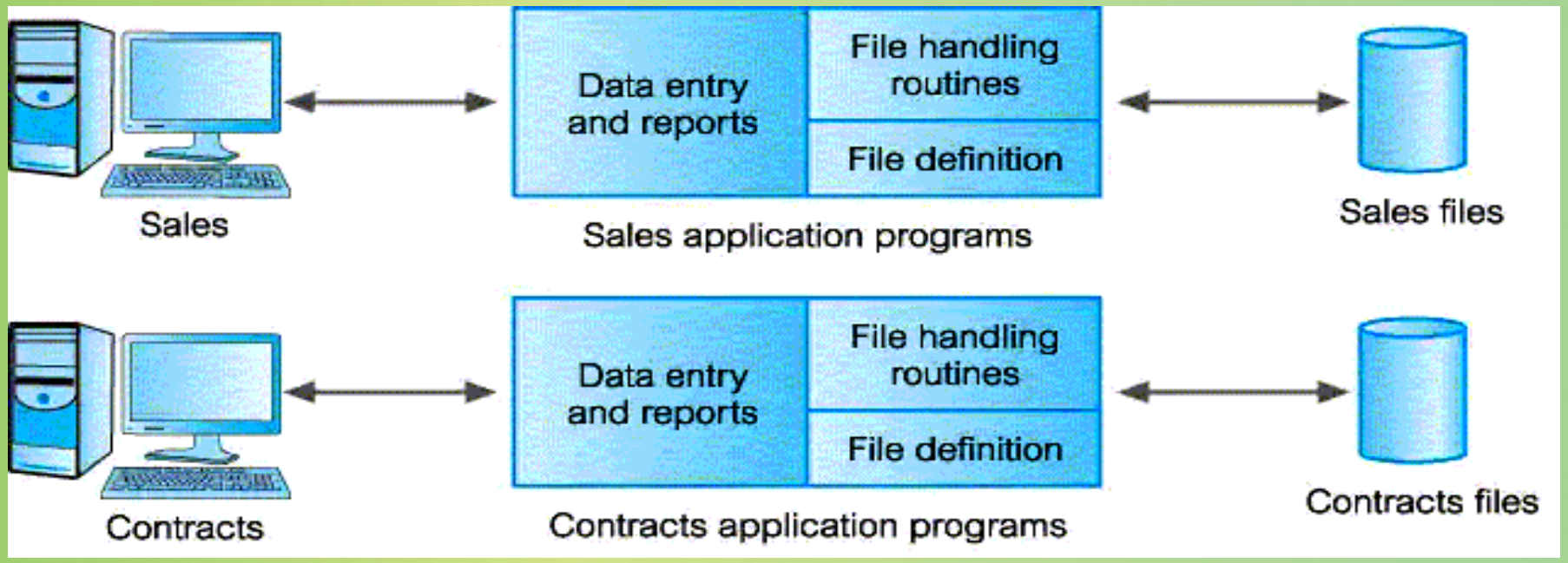
File-Based Systems

- Predecessor of the Database System
- Collection of application programs that perform services for the end users (e.g. reports).
- Each program defines and manages its own data.

DreamHome Example

- The Sales Department is responsible for the selling and renting of properties.
- A property owner wishing to rent his/her property approaches the Sales Department and fills out a form.
- A client wanting to rent property also contact the Sales Department and fills out a forms
- When a client agrees to rent a property, he/she is transferred to the Contracts Department, where lease will be signed and payment made.

File-Based Processing



Sales Files

PropertyForRent (propertyNo, street, city, postcode, type, rooms, rent, ownerNo)

PrivateOwner (ownerNo, fName, lName, address, telNo)

Client (clientNo, fName, lName, address, telNo, prefType, maxRent)

Contracts Files

Lease (leaseNo, propertyNo, clientNo, rent, paymentMethod, deposit, paid, rentStart, rentFinish, duration)

PropertyForRent (propertyNo, street, city, postcode, rent)

Client (clientNo, fName, lName, address, telNo)

Limitations of File-Based Approach

- **Separation and isolation of data**
  - Each program maintains its own set of data.
  - Users of one program may be unaware of potentially useful data held by other programs.
- **Duplication of data**
  - Same data is held by different programs.
  - Wasted space and potentially different values and/or different formats for the same item.

Limitations of File-Based Approach

- **Data dependence**
  - **File structure is defined in the program code.**
- **Incompatible file formats**
  - **Programs are written in different languages and so cannot easily access each other's files.**
- **Fixed Queries/Proliferation of application programs**
  - **Programs are written to satisfy particular functions.**
  - **Any new requirement needs a new program.**

Database Approach

- Previous limitations arose because:
  - Definition of data was embedded in application programs, rather than being stored separately and independently.
  - No control over access and manipulation of data beyond that imposed by application programs.
- Result:
  - the database and Database Management System (DBMS).

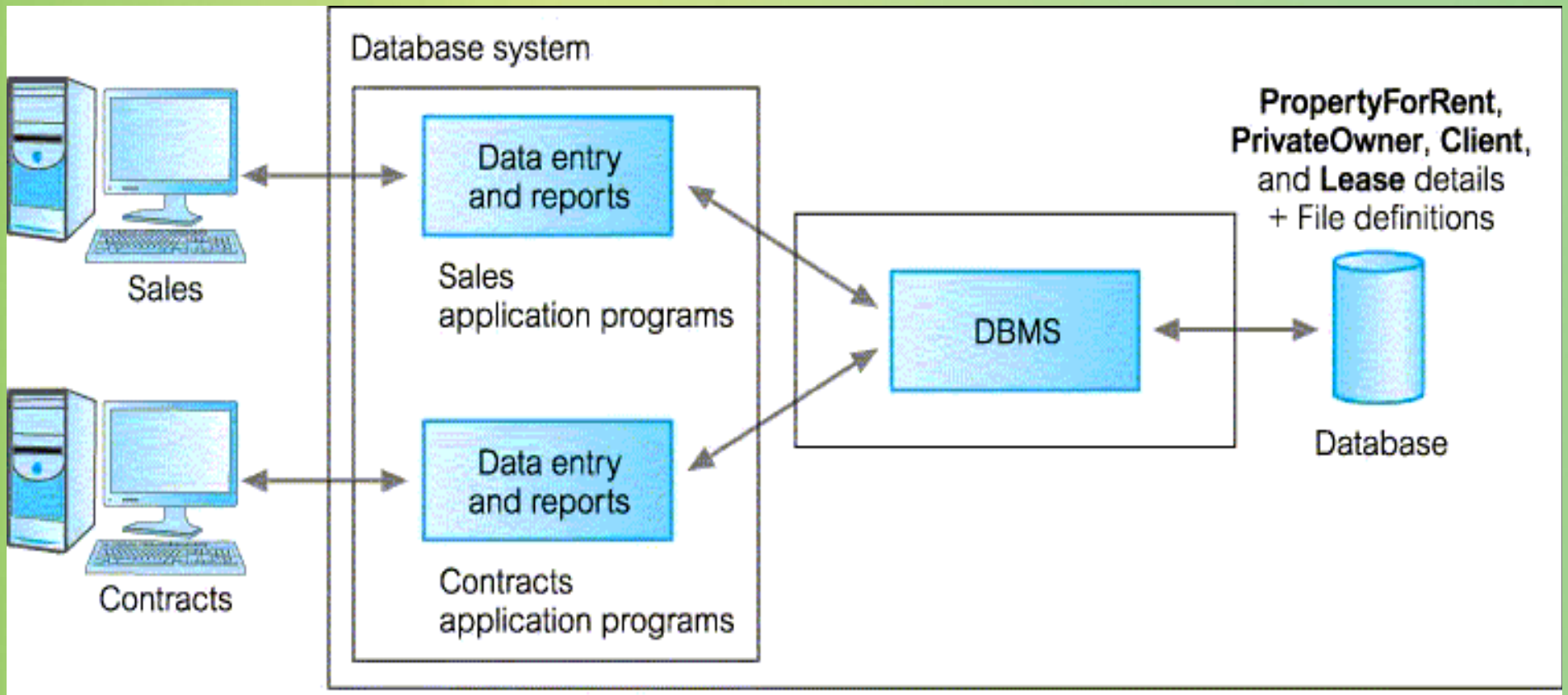
Database

- Shared collection of logically related data (and a description of this data), designed to meet the information needs of an organization.
- System catalog (metadata) provides description of data to enable program–data independence.
- Logically related data comprises entities, attributes, and relationships of an organization's information.

Database Management System (DBMS)

- A software system that enables users to define, create, maintain, and control access (insert, update, retrieve, …) to the database.
- (Database) application program: a computer program that interacts with database by issuing an appropriate request (SQL statement) to the DBMS.
  - SQL – structured query language

Database Management System (DBMS)



PropertyForRent (propertyNo, street, city, postcode, type, rooms, rent, ownerNo)
PrivateOwner (ownerNo, fName, lName, address, telNo)
Client (clientNo, fName, lName, address, telNo, prefType, maxRent)
Lease (leaseNo, propertyNo, clientNo, paymentMethod, deposit, paid, rentStart, rentFinish)

Database Approach

- DBMS supports the following features:
- Data definition language (DDL).
  - Permits specification of data types, structures and any data constraints.
  - All specifications are stored in the database.
- Data manipulation language (DML).
  - General enquiry facility (query language) of the data.

Database Approach

- Controlled access to database may include:
  - a security system – controlled access
  - an integrity system – consistency of data
  - a concurrency control system – shared access
  - a recovery control system – restore after failure
  - a user-accessible catalog – description of data

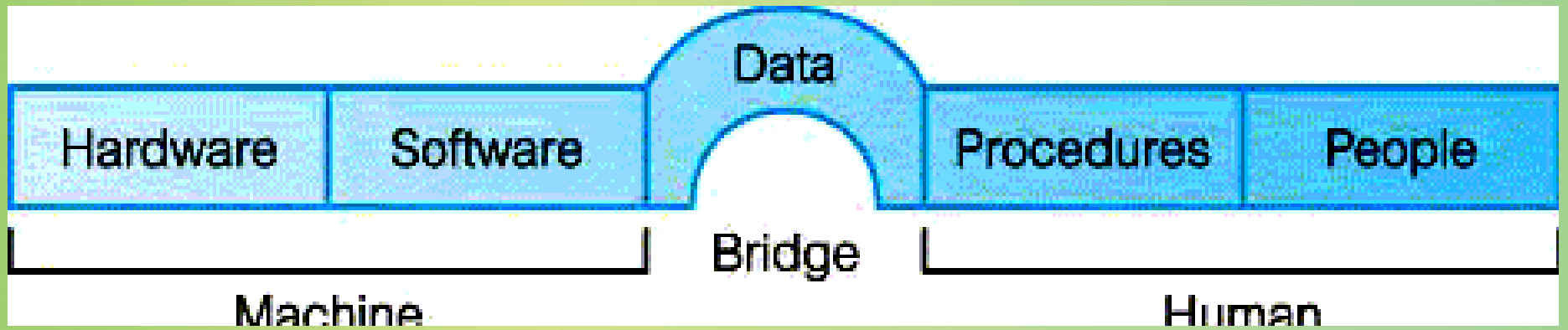
Views

- Allows each user to have his or her own view of the database.
- A view is essentially some subset of the database.

Views - Benefits

- Reduce complexity
- Provide a level of security
- Provide a mechanism to customize the appearance of the database
- Present a consistent, unchanging picture of the structure of the database, even if the underlying database is changed

Components of DBMS Environment



Components of DBMS Environment

- **Hardware**
  - **Can range from a PC to a network of computers.**
- **Software**
  - **DBMS, operating system, network software (if necessary) and also the application programs.**
- **Data**
  - **Used by the organization and a description of this data called the schema.**

Components of DBMS Environment

- **Procedures**
  - **Instructions and rules that should be applied to the design and use of the database and DBMS.**
    - **How to logon to DBMS**
    - **How to use a particular facility of DBMS**
    - **Start/stop DBMS, etc.**
- **People**
  - **People involved with the DBMS system**

Roles in the Database Environment

- **Data Administrator (DA)**
  - **Data resource management, conceptual/logical database design**
- **Database Administrator (DBA)**
  - **physical database design and implementation**
- **Database Designers (Logical and Physical)**
  - **Logical database designer is concerned with identifying the data, the relationships between the data, and the constraints on the data**
  - **Physical database designer decides how the logical database design is to be physically realized.**

Roles in the Database Environment

- **Application Programmers**
  - **Implement application programs that provide the required functionality for the end-users**
- **End Users (naive and sophisticated)**
  - **Clients of the database**

History of Database Systems

- First-generation
  - Hierarchical (IMS by IBM) and Network (IDS by GE)
  - Complexed programs…
- Second generation
  - Relational (DB2, Oracle, MySQL, SQL Server)
  - Limited modelling capability...
- Third generation
  - Object-Relational
  - Object-Oriented

Advantages of DBMSs

- Control of data redundancy
  - Data redundancy kept to minimum
- Data consistency
  - Avoid multiple copies of data
- More information from the same amount of data
  - Can access all related data
- Sharing of data
  - One database shared by everyone
- Improved data integrity
  - Constraints ensure rules for all data not violated

Advantages of DBMSs

- Improved security
  - Username, passwd, access control, …
- Enforcement of standards
  - For different departments
- Economy of scale
  - One shared database saves cost
- Balance conflicting requirements
  - DBMS can be designed to optimize the use of resources of the organization
- Improved data accessibility and responsiveness
  - Data from all departments can be accessed

Advantages of DBMSs

- Increased productivity
  - Provide many of the standard functions for application programmers
- Improved maintenance through data independence
  - Separate data descriptions from the applications
- Increased concurrency
  - Users are allowed to access the same data simultaneously
- Improved backup and recovery services
  - Provide facilities to minimize the amount of processing that is lost following a failure

Disadvantages of DBMSs

- An extremely complex piece of software
- An extremely large piece of software
- A large mainframe multi-user DBMS can be extremely expensive
- Additional storage space, larger machine, …
- Cost of conversion to a new DBMS can sometimes be higher than the DBMS
- Performance may be lower than file-based system since DBMS is general-purposed
- Higher impact of a failure since all users and applications rely on the availability of DBMS