Information Systems

Module Code: INFS-111

Programs: BCSS/BBIT

Database

A database is a collection of information that is organized so that is organized so that it can easily be accessed, managed and updated.

Database Management System (DBMS)

- DBMS is a software package that allows data to be effectively stored, retrieved and manipulated and the data stored In a DBMS package can be accessed by multiple users and by multiple application programs like (SQL Server, Oracle and MS Access)

Comparison of Traditional File-Based Approach

- File Based system were an early attempt to computerize the manual filing system. File-based system is a collection of application programs that perform services for end-user. Each program defines and manages its data.

- However problems occurred using the file based

1. Separation and Isolation of Data

When data is isolated it makes it difficult to access it.

2. Duplicate of data

When employing the decentralized file-based approach, the uncontrolled duplication of data is bound to happen

Comparison of Traditional File-Based Approach

4. Data Dependence

Making changes to an existing structure are rather difficult and will lead to a modification of program.

5. Incompatible file formats

File-Based Processing

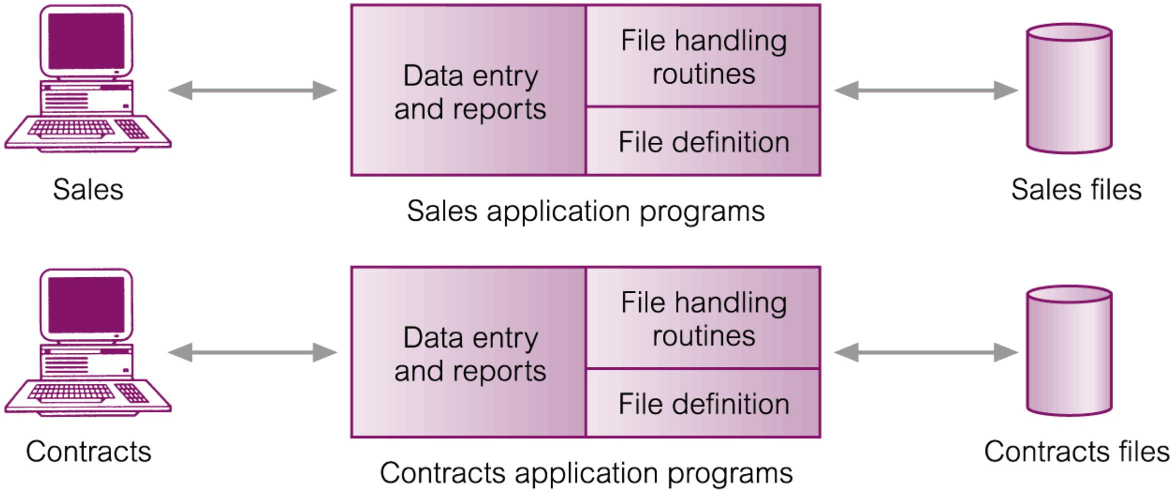


Figure 1.5
File-based processing.

Comparison of Traditional File-Based Approach

- Advantages of using Database approach

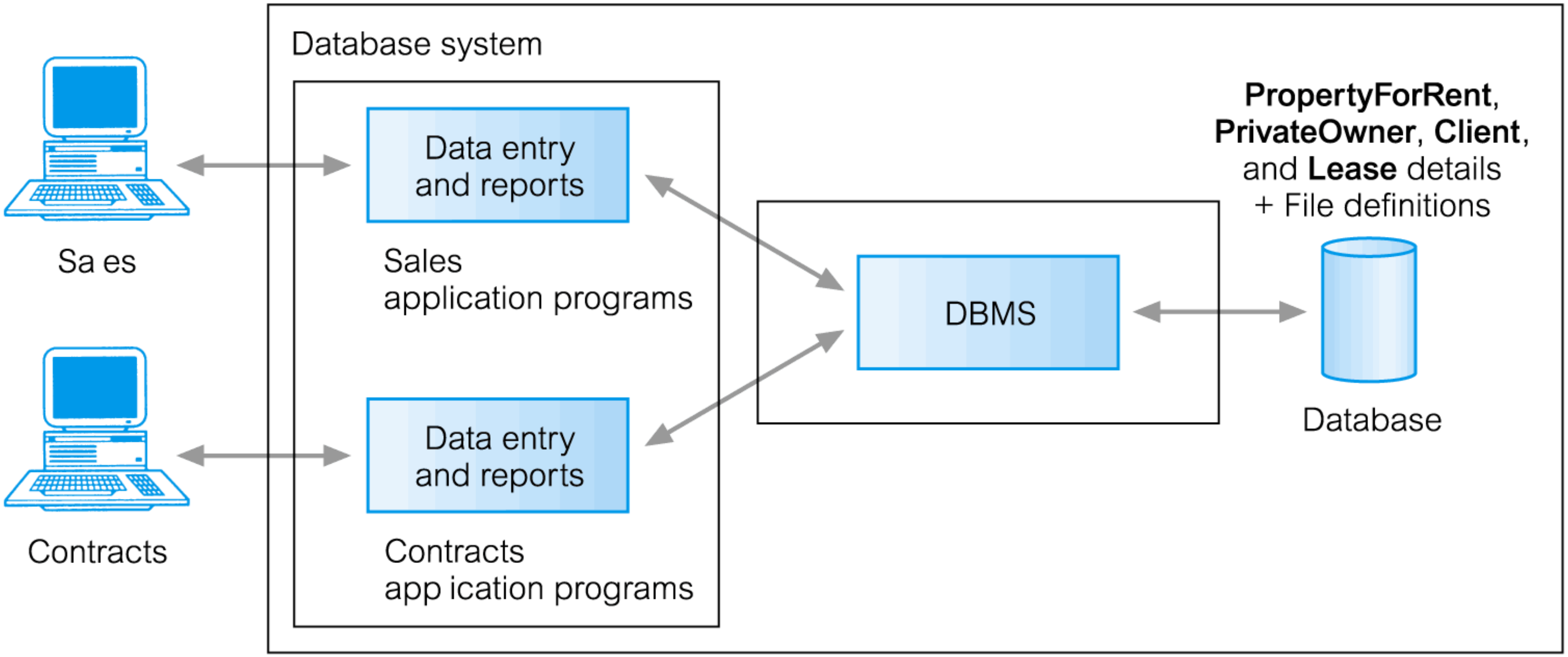
1. The centralization of data facilitates risk management: it is much easier to back-up, monitor or audit one central database than thousands of separate files.
2. The integration of data in a single logical DBMS view (although it may physically be stored in many different databases) allows for better management decision support. It is now possible to have the complete picture or profile of a customer; or to relate employee performance to sales.
3. Control of Data Redundancy
4. Data Consistency

Comparison of Traditional File-Based Approach

Disadvantages of Database approach include

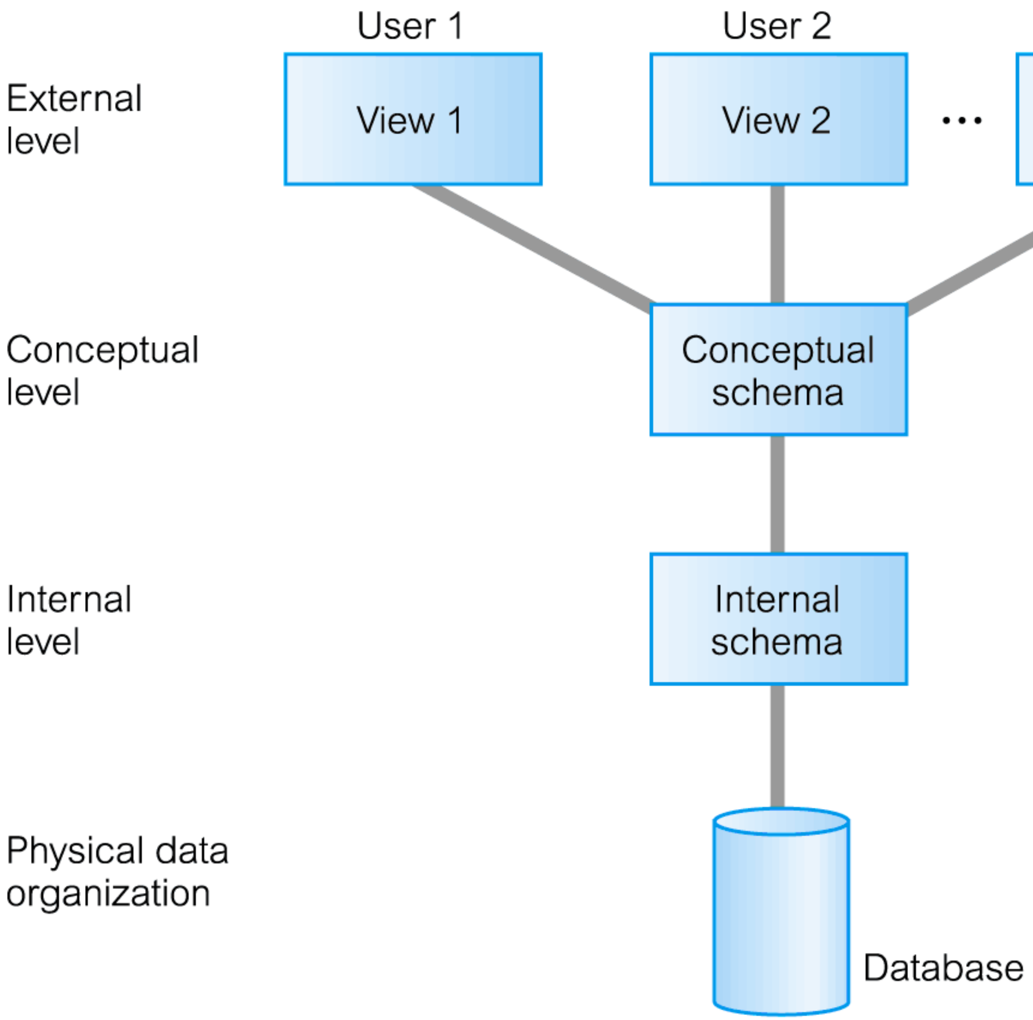
1. It introduces an additional layer of software between the application and the database, often resulting in a slower access to the data.
2. System designers and functional users loose some freedom and control over their data, since they can no longer decide exactly what and how data will be stored.
3. The reliance on one single database increases the vulnerability of the organization
4. A DBMS introduces additional costs: a typical DBMS licence is very expensive and requires specialized, highly-paid staff

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, rentF nish)

DBMS Architecture



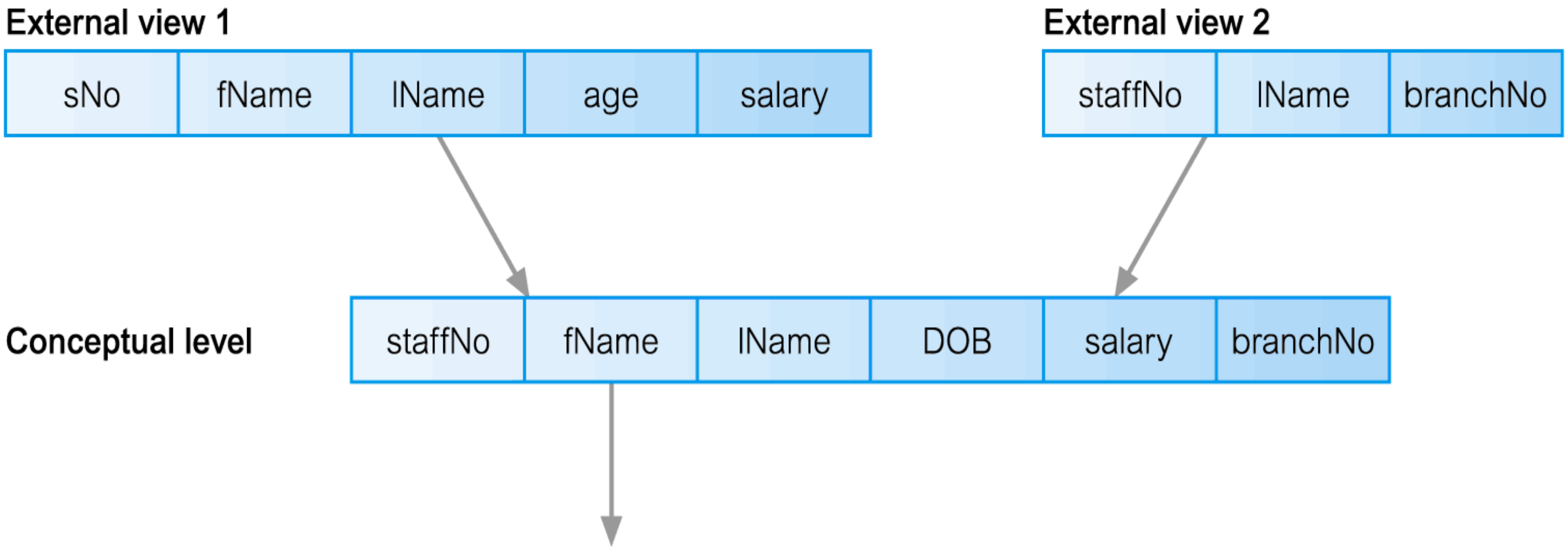
DBMS Architecture

- External Level
  - Users' view of the database.
  - Describes that part of database that is relevant to a particular user.

- Conceptual Level
  - Community view of the database.
  - Describes what data is stored in database and relationships among the data.

- Internal Level
  - Physical representation of the database on the computer.
  - Describes how the data is stored in the database

DBMS Architecture



External view 1

sNo	fName	lName	age	salary

External view 2

staffNo	lName	branchNo

Conceptual level

staffNo	fName	lName	DOB	salary	branchNo

Internal level

```
struct STAFF {
      int staffNo;
      int branchNo;
      char fName [15];
      char lName [15];
      struct date dateOfBirth;
      float salary;
      struct STAFF *next;                 /* pointer to next Staff record */
};
index staffNo; index branchNo;            /* define indexes for staff */
```

Data Structures

- Record: Within a table (or file), information will be organized in the form of records. Each record represents one instance of a real-world entity or transaction. E.g. in the customer database, there may be a record containing all the data about Joe Smith (including his address, contact numbers, credit status), there will be another record with the data for Jane Doe and so on, one record for each of the customers.
- Field: The information within a record is structured into separate fields. In the above example, Joe Smith's record would have at least four fields: his name, his address, his contact telephone number and his credit status. In practice, his name may be split into at least three separate fields: one for his surname "Smith", one for his first name "Joe", one for his designation "Mr" and probably one for his initial(s).

History of DBMS

1. **Hierarchical Data Model**
2. **Network Data Model**
3. **Relational Data Model**
4. **Object-Oriented**

Hierarchical Data Model

- The hierarchical data(base) model is the oldest and conceptually simplest model. This structure allows only for one type of relationship, the "parent-child" (or one-to-many) relationship. In addition, there can be only one single relationship between tables. This structure can also be visualised or represented using the image of a tree.

Network Data Model

- The network data model accommodates the “many-to-many” relationships often found in the real world. It allows the explicit linking of sets of entities by means of network relationships.
- In a university environment, the relationships between courses and students could not (or with great difficulty) be accommodated in a hierarchical database. Each student can enrol for one or several courses, whilst each course will have an enrolment of zero, one or, hopefully, many students.

Relational Data Model

- The relational model requires that all data is stored in data tables, with unique key fields (the primary key) and all other data fields in the table entirely and fully dependent on the key fields. In the example below, there are a large number of tables (the relational model tends to generate a large number of tables). The student table has the student number as the key field.
- Tables can be linked, by reference to their primary key fields, in any possible way. Links between tables do not have to be predefined when the database is designed, providing structural flexibility and efficient information retrieval. Each link between tables is a relationship and consists of a separate table.
- Normalisation is a step-by-step process of ensuring that a database structure conforms to the requirements of the standard relational database models

Object-Oriented

- The object-oriented (OO) approach was introduced in the chapter on software. Remember that an object encapsulates both the data and the methods or operations that can be performed on the data. Although business databases were not initially considered to be good candidates for OO, more and more applications are benefiting from the use of object-oriented database management systems (OODMBS).