1. INTRODUCTION

Background

- computer-based information systems are essential in virtually all large organisations today
- data is input manually (e.g. Banklink machine) or automatically (e.g. POS systems)

1.1. Definitions

A **database** is a *persistent* collection of *related* data supporting several *different applications* within an organisation.

Key features of definition:

- data stored in a database (DB) is *related* in some way
- DB supports (manages) data for *several* different applications
- DB is *persistent* i.e. data is permanently stored

1.1. Definitions (cont)

What is data?

- data is known facts that can be recorded and have an implicit meaning

Metadata	***Data***
Student number:	89041258
Name:	John Patrick Smith
Account balance:	42.26

A **Database Management System (DBMS)** is the set of software which manages a DB or set of DBs

1.2 DBMSs provide.....

1. Efficient, reliable and secure management of large amounts of persistent data.
2. Languages for storing, retrieving and updating data in the DB - *data manipulation languages*
3. Language(s) for defining the DB - *data definition language.*
   This data about about data (e.g. student number is a seven digit number plus one check digit) is called *metadata*

Database System (DBS)

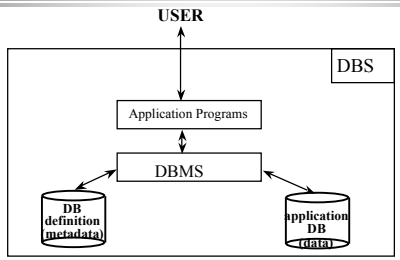

Figure 1.1

- The DBMS, together with a DB containing application data and its associated metadata, and application programs which manipulate the data, constitute a **Database System (DBS).**
- Metadata and data are stored separately

1.3 Example of a DB (GP DBS)

- records details about patient, their visits to the doctor, referrals to consultants and their accounts
- patients, accounts, referrals and consultants are all examples of real-world entities
- an ***entity*** is an "object" about which we wish to store information in the DB

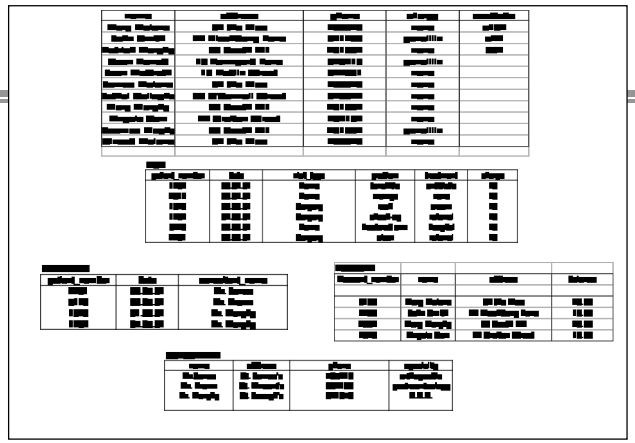

Figure 1.2

Example GP DBMS (contd.)

- entities are linked together by ***relationships***
- entities and relationships together constitute the *mini-world* (a subset of the real world) which we will *model* in the DB
- examples of relationships are:

  PATIENTs *make* VISITs to doctor
  VISITs *are charged* to ACCOUNTs
  PATIENTs *are referred to* CONSULTANTS

Example GP DBMS (contd.)

Data definition

PATIENT 'file' is divided into a set of PATIENT records, each of which stores the same information about a single patient:

- patient_number
- name
- address
- phone
- allergy

patient_number, name, address, etc. are called ***data items*** or ***fields*** or ***attributes***

- a data item is the basic unit of information stored in a DB
- other 'files' are VISIT, ACCOUNT, REFERRAL and CONSULTANT
- Figure 1.2 shows the GP DB populated with data

1.4 Data as a resource

- many organisations are critically dependent on their computer-based information systems
- data is a strategic organisational resource
- DBMSs provide the technology for managing the information resource efficiently, reliably and securely

1.5 Why database technology?

- pre-DB era (up to end of 1960's) characterised by file processing systems
- file systems offered efficient direct access to individual records and fast sequential processing
- choice of file organisation technique is based on the needs of a particular application
- where multiple applications want to share data, this can give rise to wasteful duplication (e.g. of patient names, addresses, visit charges etc.) between patient record application and accounting application

1.5 Why database technology (cont.)

- Duplication of data is wasteful of storage and inefficient but, more importantly, it leads to inconsistencies
- DB approach aims to eliminate such *redundancy* (data duplication)
- Data from all applications is integrated and stored once in the DB
- All applications access the same physical copy of the data

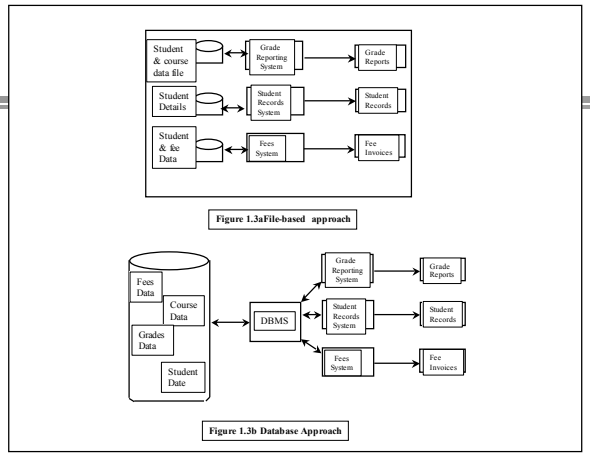

Figure 1.3aFile-based approach

Figure 1.3b Database Approach

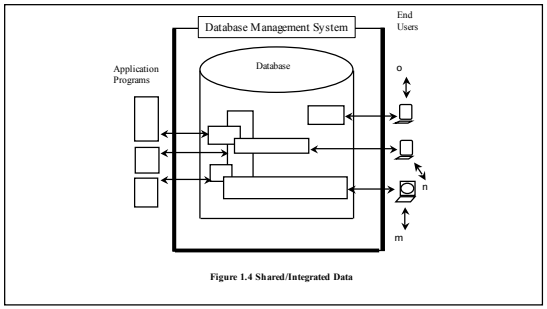

Figure 1.4 Shared/Integrated Data

Disadvantages of file based approach to data mangement:

- lack of data independence
- lack of support for data integrity
- inadequate backup and recovery mechanisms
- no query language support
- no support for management of metadata

How do DBMSs overcome these problems?

data independence

- File based systems are ***data dependent*** because the way in which data is viewed by an application ***and*** way in which it is physically stored are built in to the logic of the application program
- DBMSs support ***physical data independence*** by ***insulating*** the way in which data is viewed by the application programs/users from the way in which it is physically stored
- DBMSs support ***logical data independence*** by allowing applications to change their view of the data without affecting the physical organisation of the data

Data Integrity

- data integrity is concerned with the consistency and accuracy of the data in the DB
- data redundancy is a major threat to data integrity
- support for data integrity is a key feature of a DBMS
- DBs model a mini-world to which many rules apply
  e.g. "A student has only one address" or "A student must take 5 courses in the final year or 4 courses plus a project"
- DBMSs express such rules by means of ***integrity constraints***
- ***validation*** of data values being entered into the DB is another aspect of data integrity
- many users/application programs simultaneously updating the DB can threaten data integrity (requiring ***concurrency control***)

Backup and Recovery

- only facility available to file processing systems to restore data following failure was the back-up copy
- insufficient in many on-line environments and also organisations where data is a strategic resource
- DBMSs provide very sophisticated recovery mechanisms

Query Language Suppport

- file sytems are basically tools for physical storage of data
- file systems make data much less accessible to users than DB systems - the GP would have to write an application program every time they wanted to look up a patient record!
- DBMSs provide a variety of interfaces to suit the needs of a wide range of users

Metadata Management

- in fps, metadata is part of the application program
- leads to duplication of metadata across application programs leading to integrity problems
- a stored patient record

1234	Mary Malone	64 The Rise	290846	penicillin

Metadata Management (contd.)

- to interpret the data, we need to look at an application program:

```
Struct.patient_record    {
        short int     patient_number;
        char          name          [30];
        char          address       [40];
        char          phone         [6];
        char          allergy       [10];
                                        };
```


Metadata Management (contd.)

With DB approach:

- •metadata is stored centrally in the *catalogue*
- •DB catolog entry for patient record
- •Patient_record contains basic details on patient

patient_number	alpha (4)	unique;
name	alpha (30)	firstname followed by surname;
address	alpha (40)	truncate if necessary:
phone	alpha (6)	home phone;
allergy	alpha (10)	drug name or none;

Disadvantages of the DB approach

- cost of software, hardware and training
- loss of autonomy brought about by centralised control of the data
- inflexibility due to complexity

Database Users

Database Administrator (DBA)

- has overall responsibility for the DB including deciding on information content, specifying access constraints, selection of appropriate backup and recovery measures, monitoring performance etc.

Other users of the DB:

- systems analysts
- DB designers