Chapter 1
Database and Database Users

What Is a Database?

3 Logically coherent collection of data with inherent meaning.

3 A database is an integrated, shared repository of operational data of interest to an enterprise.

Integration means it must be the unification of several distinct files

(IBM-AS/400, Ingres, Oracle, Borland’s Paradox, MS-Access, MS SQL/Server MS-FoxPro Informix and dBASE are integrated.

; MS Works database facilities and Lotus 123’s database features are not. (they work with one file at a time!)

Shared means the same data can be used by more than one user (concurrently?)

(AS/400, Oracle, Ingres, MS-FoxPro, Informix MS SQL/Server, and MS-Access are shared.

; dBASE, MS Works and Lotus are not shared.)
Database Management System (DBMS):

3 A collection of programs that enable the user to create and maintain a database.

3 A general purpose software system that facilitates the following:

1) Defining the database (schema, define the attributes, their types, size, constraints, range, etc.)

2) Populate the database (Add data to the database)

3) Manipulate the database (Querying, updating, generating reports, creating views, joining information)

3 A computer program which manages access to a database:

- PROGRAM is a system program. Typically not an application program

- ACCESS means create, retrieve, insert, update, delete (Hides the hardware details)
What is a **Database System**:

3 The combination of DBMS and application programs written to access and manipulate the database is called a database system.

See Figure 1.1 page 3
What is Metadata?

3 Metadata is non-user data which contains the descriptive information of the data and database organization.

3 Data about data!

Class discussion:

Why maintain meta-data?

How would you maintain this information?

What are the implementation issues?

How does this affect our applications? (441 students?)
Why Use a Database?

1) Compactness (no paper files)

2) Better access to data, more currency (up-to-date info available at command)

3) Centralized control of operational data (by DBA)

- Redundancy can be reduced and controlled
- Data shared (concurrent use must be controlled)
- Standards enforced (corporate, department, industry, national, international)
- Integrity maintained (accuracy) system wide semantic constraint maintained, inconsistency is avoided

Example:
- (if male, no maiden name)
- (If middle_manager, Salary < 50k)
- (if customer is to be deleted, then there must be no outstanding invoices!)

3) Conflicting requirements are identified and balanced.
   (DBA can optimize the database for the whole company)

4) Data Independence (really an objective rather than an advantage):

Data Dependence = knowledge of stored data organization & access technique are built into application logic and code. Can't change storage structure w/o affecting code.

Data Independence = application's immunity to storage structure & access strategy.
Characteristics distinguishing a database system from a file system:

1. Existence of a catalog or data dictionary
   C Meta-data
   C Self Describing

2. Data abstraction
   C The details about the data are hidden from user

3. Program-data and Program-operation independence

4. Support of multiple user views

5. Sharing of data among multiple transactions

Main categories of database users are:

1. Database designers and administrators

2. End-users

3. Application programmers and system analysts
What should the DBMS provide its user community?

1. Controlled redundancy

2. Restricted and authorized access
   - User level
   - File level
   - Record Level
   - Attribute level

3. Persistent storage

4. Inference capability (use of deductive rules) (predicate Logic, Prolog)

5. Multiple interfaces (programmatic, graphical, command line)

6. Represent complex relationships among data

7. Enforce integrity constraints

8. Provide backup and recovery

9. Provide OLTP (online transaction processing)

10. Provide OLAP (online analytical processing)
Advantages of database approach over traditional file processing systems:

1. **Potential for enforcing standards**
   * One company, one DBA, one standard! (hopefully)

2. **Flexibility**
   * Modifying the organization of data is easy.

3. **Reduced application development time**
   * Application programmers don't have to have expertise in file organization.

4. **Availability of up-to-date information to all users**
   * Allows concurrent multi-user access

5. **Economies of scale**
   * One integrated database system is cheaper than many file processing system in each department!
   * Reduces the overall cost of operation and management of data)
When not to use a DBMS:

1. Initial investment in hardware, software and training is prohibitively expensive.

2. No need for the generality that a DBMS provides for defining and processing data.

3. Overhead for providing security, concurrency control, recovery and integrity is too high for the application at hand.

4. The database and its applications are simple, well-defined and not expected to change.

5. There are stringent real-time requirements for some programs that may not be met because of DBMS overhead.

6. Multiple user access to data is not required (nor will ever be)