Chapter 1

- Objectives: to learn
- The difference between data and information
- What a database is, the various types of databases, and why they are valuable assets for decision making
- The importance of database design
- How modern databases evolved from file systems
- About flaws in file system data management
- The main components of the database system
- The main functions of a database management system (DBMS)

Introduction

- Good decisions require good information derived from raw facts
- Data is managed more efficiently when stored in a database
- Databases solve many of the problems encountered in data management
  - Used in most modern settings involving data :
    - Business, Research, Administration
- Important to understand how databases work and interact with other applications
- Data Management:
  - the discipline of generating, storing, & retrieving data
  - A "Core" activity of any organization

Data vs. Information

- **Data** are raw facts
  - **Information** is the result of processing raw data to reveal meaning
  - Information requires context to reveal meaning
  - Raw data must be formatted for storage, processing, and presentation
- **Data**: building blocks of information
  - Information produced by processing data
  - Information used to reveal meaning in data
  - Accurate, relevant, timely information is the key to good decision making
  - Good decision making is the key to organizational survival

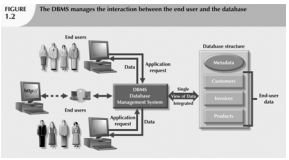
Introducing the Database

- Database: shared, integrated computer structure that stores a collection of:
  - End-user data: raw facts of interest to end user
  - **Metadata**: data about data
    - Provides description of data characteristics and relationships in data
    - Complements and expands value of data
- **Database management system (DBMS)**: collection of programs
  - Manages structure and controls access to data

Role and Advantages of the DBMS

- DBMS is the intermediary between the user and the database
  - Database structure stored as file collection
  - Can only access files through the DBMS
- DBMS enables data to be shared
- DBMS integrates many users' views of the data

FIGURE 1.2 The DBMS manages the interaction between the end user and the database



Role and Advantages of the DBMS

- Advantages of a DBMS:
  - Improved data sharing
  - Improved data security
  - Better data integration
  - Minimized **data inconsistency**
- Results in:
  - Improved data access
  - Improved decision making
  - Increased end-user productivity as it
    - Promotes integrated view of organization's operations
    - Reduces the probability of data inconsistency
    - enables producing quick answers to ad hoc queries

Types of Databases

- **Databases** can be classified according to:
  1. Number of users
  2. Database location(s)
  3. Expected type and extent of use
- **Single-user database** supports only one user at a time
  - Desktop database: single-user; runs on PC
- **Multiuser database** supports multiple users at the same time
  - **Workgroup**
  - **Enterprise**

Types of Databases

- **Centralized database**: data located at one site
- **Distributed database**: data distributed across several different sites
- **Operational database**: supports a company's day-to-day operations
  - Transactional or production database
- **Data warehouse**: stores data used for tactical or strategic decisions

TABLE 1.1 Types of Databases

PRODUCT	NUMBER OF USERS			DATA LOCATION		DATA USAGE		XML
	SINGLE USER	MULTIUSER		CENTRALIZED	DISTRIBUTED	OPERATIONAL	DATA WAREHOUSE	
		WORKGROUP	ENTERPRISE					
MS Access	X	X		X		X		
MS SQL Server	X[1]	X	X	X	X	X	X	X
IBM DB2	X[1]	X	X	X	X	X	X	X
MySQL	X	X	X	X	X	X	X	X*
Oracle RDBMS	X[1]	X	X	X	X	X	X	X

* Supports XML functions only. XML data are stored in large text objects.

Types of Data in Databases

- **Unstructured data** exist in their original state
- **Structured data** result from formatting
  - Structure applied based on type of processing to be performed
- **Semistructured data** have been processed to some extent
- **Extensible Markup Language (XML)** represents data elements in textual format
  - **XML database** supports semistructured XML data

Why Database Design Is Important

- **Database design** focuses on design of database structure used for end-user data
  - Designer must identify database's expected use
- Well-designed database:
  - Facilitates data management
  - Generates accurate and valuable information
- Poorly designed database:
  - Causes difficult-to-trace errors
  - Ultimately can lead to poor decisions and failure of an organization.

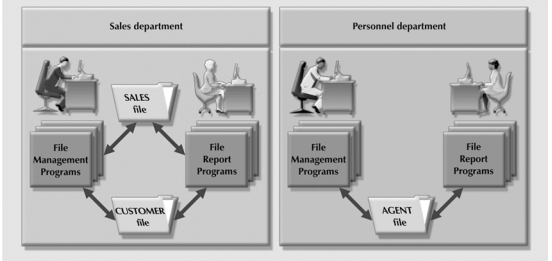
Evolution of File Systems

- Reasons for studying file systems:
  - Complexity of database design is easier to understand
  - Understanding file system problems helps to avoid problems with DBMS systems
  - Knowledge of file system is useful for converting file system to database system
- File systems typically composed of collection of file folders, each tagged and kept in cabinet
  - Organized by expected use
- Contents of each file folder are logically related
  - Manual systems were data repositories for small data collections & cumbersome for large collections

Evolution of File Systems

- Computerized file systems
  - **Data processing (DP) specialist** converted computer file structure from manual system
    - Wrote software that managed the data
    - Designed the application programs
  - Initially, computer file systems resembled manual systems
- As number of files increased, file systems evolved
  - Each file used its own application program to store, retrieve, and modify data
  - Each file was owned by individual or department that commissioned its creation

FIGURE 1.5 A simple file system



Basic File Terminology

TABLE 1.2 Basic File Terminology

TERM	DEFINITION
Data	"Raw" facts, such as a telephone number, a birth date, a customer name, and a year-to-date (YTD) sales value. Data have little meaning unless they have been organized in some logical manner.
Field	A character or group of characters (alphabetic or numeric) that has a specific meaning. A field is used to define and store data.
Record	A logically connected set of one or more fields that describes a person, place, or thing. For example, the fields that constitute a record for a customer might consist of the customer's name, address, phone number, date of birth, credit limit, and unpaid balance.
File	A collection of related records. For example, a file might contain data about the students currently enrolled at Gigantic University.

FIGURE 1.3 Contents of the CUSTOMER file

C_NAME	C_PHONE	C_ADDRESS	C_ZIP	A_NAME	A_PHONE	TP	AMT	REN
Alfred A. Ramas	615-844-2573	218 Fork Rd., Baton, TN	36123	Leah F. Hahn	615-882-1244	T1	100.00	05-Apr-2010
Leona K. Dunne	713-894-1238	Box 12A, Fox, KY	25246	Alex B. Alby	713-228-1249	T1	250.00	16-Jun-2010
Kathy W. Smith	615-894-2285	125 Oak Ln., Baton, TN	36123	Leah F. Hahn	615-882-2144	S2	150.00	29-Jan-2011
Paul F. Olowski	615-894-2180	217 Lee Ln., Baton, TN	36123	Leah F. Hahn	615-882-1244	S1	300.00	14-Oct-2010
Myron Orlando	615-222-1672	Box 111, Nino, TN	36155	Alex B. Alby	713-228-1249	T1	100.00	20-Dec-2010
Amy B. O'Brien	713-442-3381	387 Troll Dr., Fox, KY	25246	John T. Okon	615-123-5589	T2	800.00	22-Sep-2010
James G. Brown	615-297-1228	21 Tye Rd., Nash, TN	37118	Leah F. Hahn	615-882-1244	S1	120.00	25-Mar-2011
George Williams	615-290-2556	155 Maple, Nash, TN	37119	John T. Okon	615-123-5589	S1	250.00	17-Jul-2010
Anne G. Farriss	713-382-7185	2119 Elm, Crew, KY	25432	Alex B. Alby	713-228-1249	T2	100.00	03-Dec-2010
Olette K. Smith	615-297-3809	2702 Main, Nash, TN	37118	John T. Okon	615-123-5589	S2	500.00	14-Mar-2011

C_NAME = Customer name
C_PHONE = Customer phone
C_ADDRESS = Customer address
C_ZIP = Customer zip code
A_NAME = Agent name
A_PHONE = Agent phone
TP = Insurance type
AMT = Insurance policy amount, in thousands of $
REN = Insurance renewal date

Problems with File Systems

- File systems were an improvement over manual system
  - File systems used for more than two decades
  - Understanding the shortcomings of file systems aids in development of modern databases
  - Many problems not unique to file systems
- Even simple file system retrieval task required extensive programming
  - Ad hoc queries impossible
  - Changing existing structure difficult

Problems with File Systems

- Security features difficult to program
  - Often omitted in file system environments
- Summary of file system limitations:
  - Requires extensive programming
  - Cannot perform ad hoc queries
  - System administration is complex and difficult
  - Difficult to make changes to existing structures
  - Security features are likely to be inadequate

Structural and Data Dependence

- **Structural dependence**: access to a file is dependent on its own structure
  - All file system programs must be modified to conform to a new file structure
- **Structural independence**: change file structure without affecting data access
- **Data dependence**: access to data is dependent on the data characteristics
  - All file system programs must be modified to conform to new data characteristics
- **Data independence**: data storage characteristics do not affect data access

Structural and Data Dependence

- Practical significance of data dependence is difference between logical and physical format
- **Logical data format**: how human views the data
- **Physical data format**: how computer must work with data
- Each program must contain:
  - Lines specifying opening of specific file type
  - Record specification
  - Field definitions

Data Redundancy

- File system structure makes it difficult to combine data from multiple sources
  - Vulnerable to security breaches
- Organizational structure promotes storage of same data in different locations
  - Islands of information
- Data stored in different locations is unlikely to be updated consistently
- **Data redundancy**: same data stored unnecessarily in different places

Effects of Data Redundancy

- **Data inconsistency**: different and conflicting versions of same data occur at different places
- **Data anomalies**: abnormalities when all changes in redundant data are not made correctly
  - Update anomalies
  - Insertion anomalies
  - Deletion anomalies

Lack of Design and Data-Modeling Skills

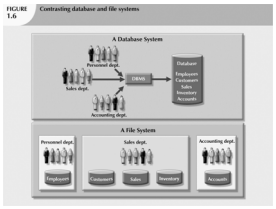
- Most users lack the skill to properly design databases, despite multiple personal productivity tools being available
- Data-modeling skills are vital in the data design process
- Good data modeling facilitates communication between the designer, user, and the developer

Database Systems

- Database system consists of logically related data stored in a single logical data repository
  - May be physically distributed among multiple storage facilities
  - DBMS eliminates most of file system's problems
  - Current generation stores data structures, relationships between structures, and access paths
    - Also defines, stores, and manages all access paths and components

The Database System Environment

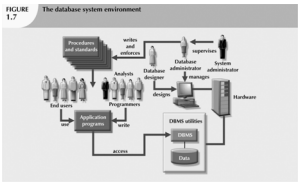
- **Database system**: defines and regulates the collection, storage, management, use of data
- Five major parts of a database system:
  - Hardware
  - Software
  - People
  - Procedures
  - Data



The Database System Environment

- **Hardware**: all the system's physical devices
- **Software**: three types of software required:
  - Operating system software
  - DBMS software
  - Application programs and utility software
- **People**: (System and database administrators, Database designers, Systems analysts and programmers, End users
- **Procedures**: instructions and rules that govern the design and use of the database system
- **Data**: the collection of facts stored in the database

The Database System Environment



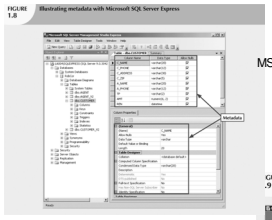
- **Database systems** are created and managed at different levels of complexity
- Database solutions must be cost-effective as well as tactically and strategically effective
- Database technology already in use affects selection of a database system

DBMS Functions

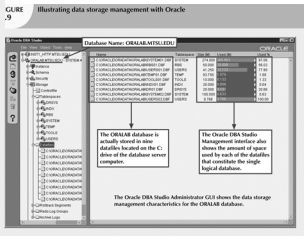
- Most functions are transparent to end users
  - Can only be achieved through the DBMS
- Data dictionary management
  - DBMS stores definitions of data elements and relationships (metadata) in a **data dictionary**
  - DBMS looks up required data component structures and relationships
  - Changes automatically recorded in the dictionary
  - DBMS provides data abstraction and removes structural and data dependency

DBMS Functions

- Data storage management
  - DBMS creates and manages complex structures required for data storage
  - Also stores related data entry forms, screen definitions, report definitions, etc.
  - DBMS stores the database in multiple physical data files
  - **Performance tuning**: activities that make the database perform more efficiently



MS Access Metadata



Oracle data storage management

DBMS Functions

- Data transformation and presentation
  - DBMS transforms data entered to conform to required data structures
  - DBMS transforms physically retrieved data to conform to user's logical expectations
- Security management
  - DBMS creates a security system that enforces user security and data privacy
  - Security rules determine which users can access the database, which items can be accessed, etc.
- Multiuser access control
  - DBMS uses sophisticated algorithms to ensure concurrent access does not affect integrity

DBMS Functions

- Backup and recovery management
  - DBMS provides backup and data recovery to ensure data safety and integrity
  - Recovery management deals with recovery of database after a failure
    - Critical to preserving database's integrity
- Data integrity management
  - DBMS promotes and enforces integrity rules
    - Minimizes redundancy
    - Maximizes consistency
  - Data relationships stored in data dictionary used to enforce data integrity
  - Integrity is especially important in transaction-oriented database systems

DBMS Functions

- Database access languages and application programming interfaces
  - DBMS provides access through a query language
  - **Query language** is a nonprocedural language
  - **Structured Query Language (SQL)** is the de facto query language for relational databases
    - ANSII standard supported by majority of DBMS vendors

DBMS Functions

- Database communication interfaces
  - Current DBMSs accept end-user requests via multiple different network environments
  - Communications accomplished in several ways:
    - End users generate answers to queries by filling in screen forms through Web browser
    - DBMS automatically publishes predefined reports on a Web site
    - DBMS connects to third-party systems to distribute information via e-mail

Managing the Database System: A Shift in Focus

- Database system provides a framework in which strict procedures and standards enforced
  - Role of the IT professional changes from programming to managing organization's resources
- Database system enables more sophisticated use of the data
- Data structures created within the database and their relationships determine effectiveness
- Disadvantages of database systems:
  - Increased costs, Management complexity, Maintaining currency, Vendor dependence, Frequent upgrade/ replacement cycles

Summary

- Data are raw facts
- Information is the result of processing data to reveal its meaning
- Accurate, relevant, and timely information is the key to good decision making
- Data are usually stored in a database
- DBMS implements a database and manages its contents
- Metadata is data about data
- Databases evolved from manual and computerized file systems

Summary

- Database design defines the database structure
  - Well-designed database facilitates data management and generates valuable information
  - Poorly designed database leads to bad decision making and organizational failure
- In a file system, data stored in independent files
  - Each requires its own management program
- Database management systems were developed to address file system's inherent weaknesses
- Some limitations of file system data management:
  - Requires extensive programming
  - System administration is complex and difficult

Summary

- More limitations of file system data management:
  - Changing existing structures is difficult
  - Security features are likely inadequate
  - Independent files tend to contain redundant data
    - Structural and data dependency problems
- DBMS present database to end user as single repository
  - Promotes data sharing
  - Eliminates islands of information
- DBMS enforces data integrity, eliminates redundancy, and promotes security