DATABASE SYSTEMS

DESIGN IMPLEMENTATION AND MANAGEMENT

INTERNATIONAL EDITION

ROB • CORONEL • CROCKETT

CHAPTER 1

DATABASE SYSTEMS

In this chapter, you will learn:

- The difference between data and information
- What a database is, what the different types of databases are, and why they are valuable assets for decision making
- The importance of database design
- How modern databases evolved from file systems

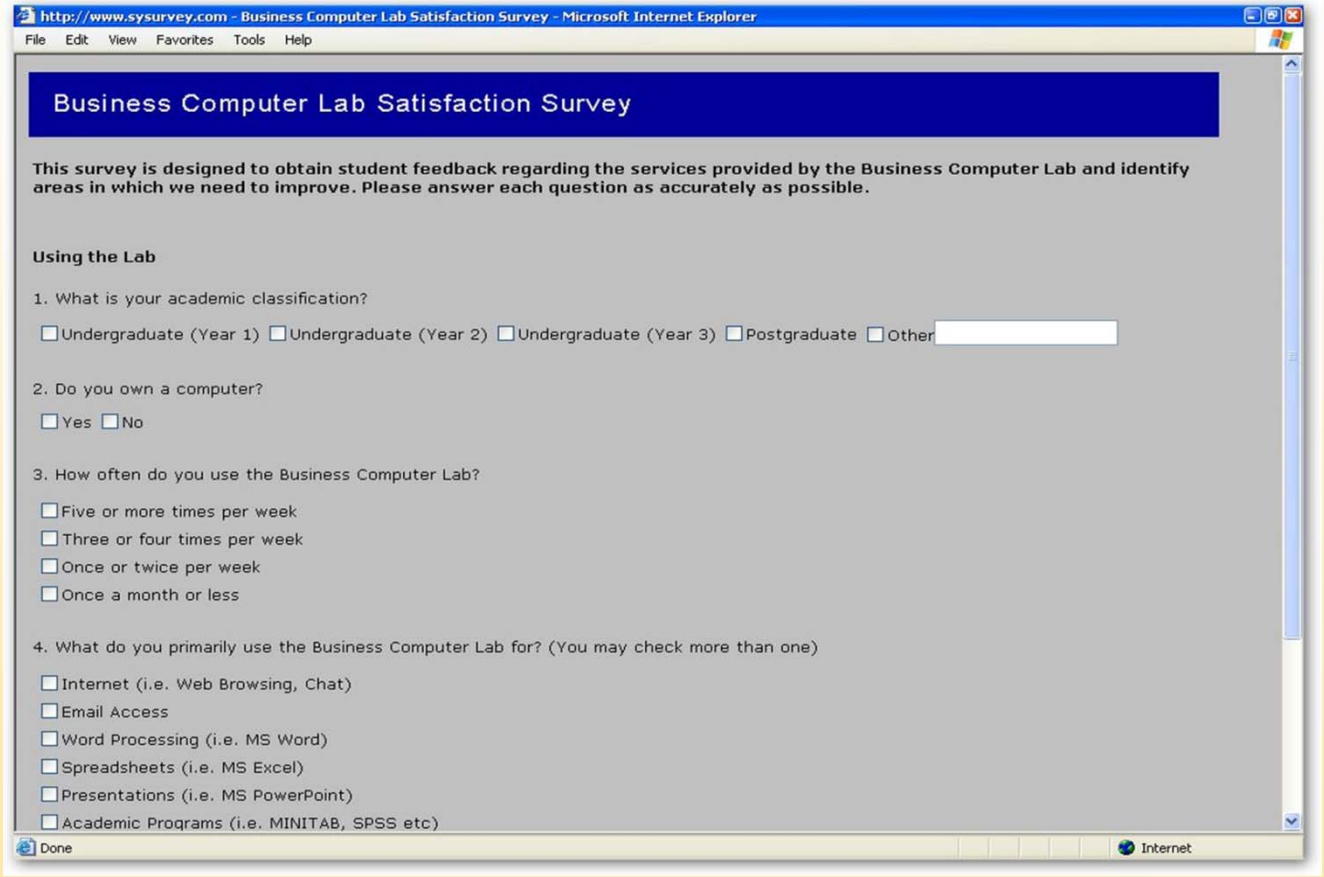
In this chapter, you will learn (continued):

- About flaws in file system data management
- What the database system's main components are and how a database system differs from a file system
- The main functions of a database management system (DBMS)

Data vs. Information

- Data:
  - Raw facts; building blocks of information
  - Unprocessed information
- Information:
  - Data processed to reveal meaning
- Accurate, relevant, and timely information is key to good decision making
- Good decision making is the key to survival in a global environment

Transforming Raw Data into Information

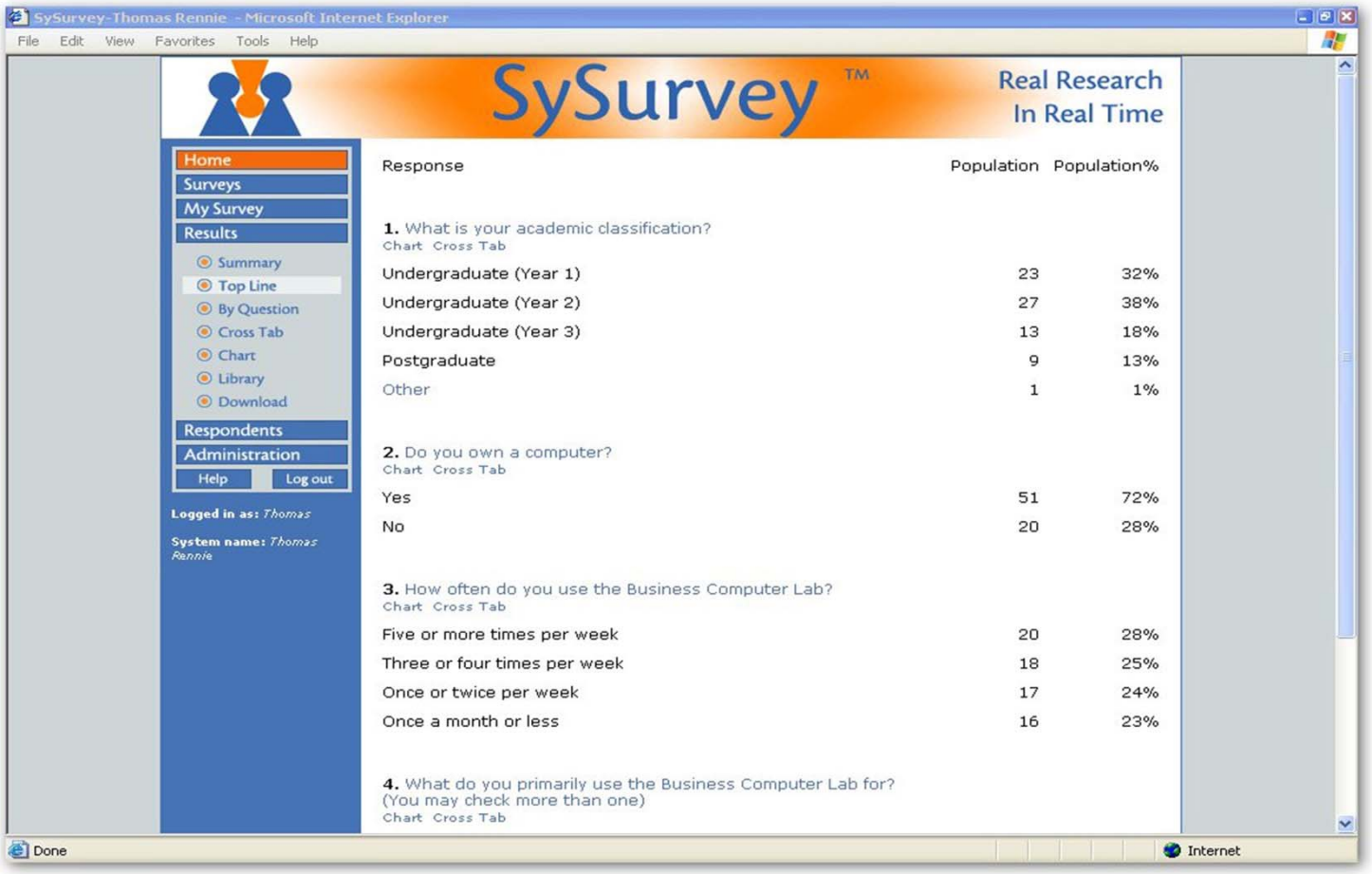


Transforming Raw Data into Information (continued)

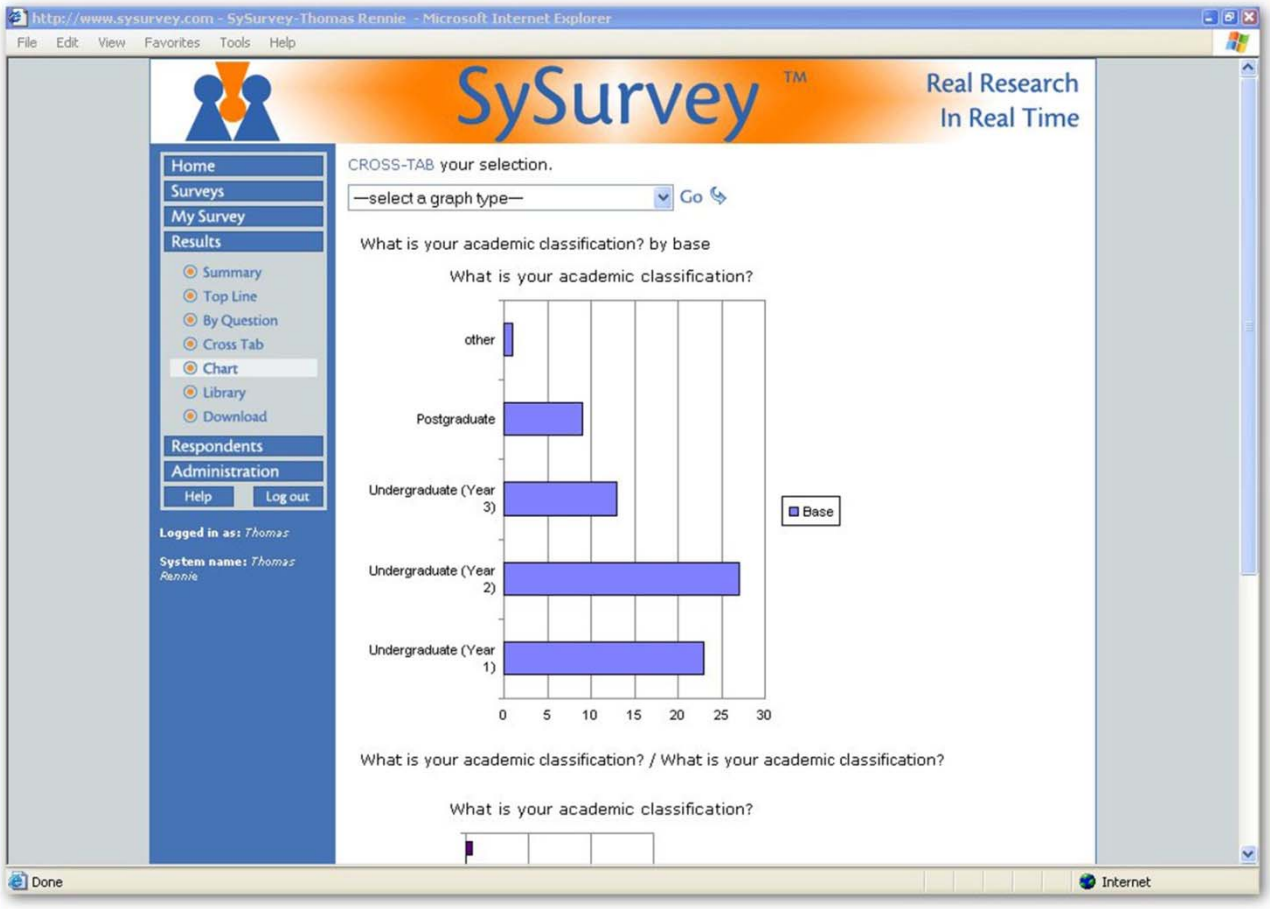
	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	
1	1-Undergra	1-Undergra	1-Undergra	1-Postgrac	1-Other	2-Yes	2-No	3-Five or m	3-Three or	3-Once or	3-Once a r	4-Internet i	4-Email Ac	4-Word Pr	4-Spreadsl	4-F
2		1					1		1				1			
3	1		1				1	1				1		1		
4	1					1			1							
5	1					1		1						1	1	
6	1						1				1					
7				1		1			1			1			1	
8		1				1			1				1			
9	1					1			1							
10				1			1				1				1	
11		1				1			1				1			
12		1				1					1	1				
13		1				1				1			1	1	1	
14				1		1				1				1		
15		1				1				1		1	1	1	1	
16	1						1		1							
17		1				1		1				1	1			
18			1			1			1						1	
19		1					1	1					1	1		
20			1			1		1				1				
21	1					1				1						
22		1					1			1						
23	1					1				1			1		1	
24		1				1			1				1			
25	1					1					1					
26			1			1		1							1	
27	1					1		1								
28	1						1			1					1	
29				1			1		1				1			
30	1			1		1					1			1		
31	1						1				1	1			1	
32					PHD	1			1			1				
33	1					1				1		1	1	1	1	
34		1				1			1			1		1	1	
35	1						1	1				1	1	1		
36				1		1					1		1			
37	1					1			1			1			1	

CSVAR005235a

Transforming Raw Data into Information (continued)



Transforming Raw Data into Information (continued)



Introducing the Database and the DBMS

- Database—shared, integrated computer structure that stores:
  - End user data (raw facts)
  - Metadata (data about data)

Introducing the Database and the DBMS (continued)

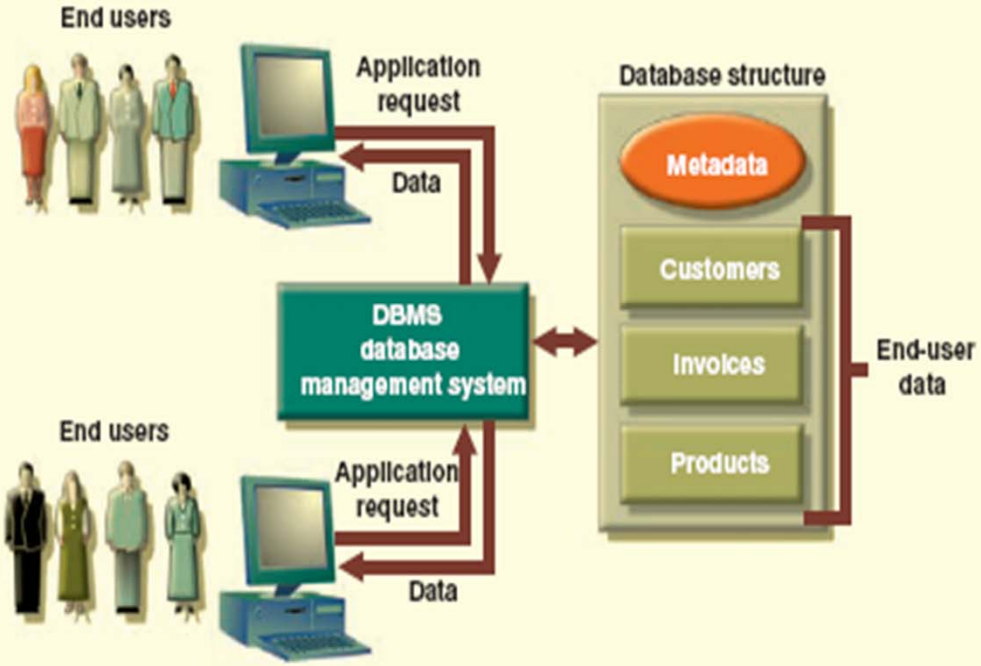
- DBMS (database management system):
  - Collection of programs that manages database structure and controls access to data
  - Possible to share data among multiple applications or users
  - Makes data management more efficient and effective

Role and Advantages of the DBMS (continued)

- End users have better access to more and better-managed data
  - Promotes integrated view of organization's operations
  - Probability of data inconsistency is greatly reduced
  - Possible to produce quick answers to ad hoc queries

Role and Advantages of the DBMS (continued)

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



Types of Databases

- Single-user:
  - Supports only one user at a time
- Desktop:
  - Single-user database running on a personal computer
- Multi-user:
  - Supports multiple users at the same time

Types of Databases (continued)

- Workgroup:
  - Multi-user database that supports a small group of users or a single department
- Enterprise:
  - Multi-user database that supports a large group of users or an entire organization

Types of Databases (continued)

Can be classified by location:

- Centralized:
  - Supports data located at a single site
- Distributed:
  - Supports data distributed across several sites

Types of Databases (continued)

Can be classified by use:

- Transactional (or production):
  - Supports a company's day-to-day operations
- Data warehouse:
  - Stores data used to generate information required to make tactical or strategic decisions
  - Often used to store historical data
  - Structure is quite different

Why Database Design is Important

- Defines the database's expected use
- Different approach needed for different types of databases
- Avoid redundant data
- Poorly designed database generates errors → leads to bad decisions → can lead to failure of organization

Historical Roots: Files and File Systems

- Managing data with file systems is obsolete
  - Understanding file system characteristics makes database design easier to understand
  - Awareness of problems with file systems helps prevent similar problems in DBMS
  - Knowledge of file systems is helpful if you plan to convert an obsolete file system to a DBMS

Historical Roots: Files and File Systems (continued)

Manual File systems:

- Collection of file folders kept in file cabinet
- Organization within folders based on data's expected use (ideally logically related)
- System adequate for small amounts of data with few reporting requirements
- Finding and using data in growing collections of file folders became time-consuming and cumbersome

Historical Roots: Files and File Systems (continued)

Conversion from manual to computer system:

- Could be technically complex, requiring hiring of data processing (DP) specialists
- Resulted in numerous "home-grown" systems being created
- Initially, computer files were similar in design to manual files (see Figure 1.3)

Historical Roots: Files and File Systems (continued)

C_NAME	C_PHONE	C_ADDRESS	C_POSTCODE	A_NAME	A_PHONE	TP	AMT	REN
Alfred A. Ramas	32-3-8891367	Stationsplein 2, Bornem 2880 Bornem	2880	Leah F. Hahn	27-21-410-7100	T1	€100.00	05-Apr-2007
Leona K. Dunne	0181-894-1238	Box 12A Rd, Highgate, London	N6 4WE	Alex B. Alby	0161-228-1249	T1	€250.00	16-Jun-2007
Kathy W. Smith	32-3-8890340	Rijksweg 58, Purrs Purrs	2880	Leah F. Hahn	27-21-410-7100	S2	€150.00	29-Jan-2008
Paul F. Olowski	31-20-6226060	Professor Tulpplein 1, Amsterdam	1018	Leah F. Hahn	27-21-410-7100	S1	€300.00	14-Oct-2007
Myron Orlando	0161-222-1672	Box 111 Dr., Rusholme, Manchester	M15 REE	Alex B. Alby	0181-228-1249	T1	€100.00	28-Dec-2007
Amy B. O'Brian	0181-442-3381	387 Troll Dr., Highgate, London	N6 LOP	John T. Okon	0181-123-5589	T2	€850.00	22-Sep-2007
James G. Brown	33-5-59200506	68 Boulevard du Général, Paris	647000	Leah F. Hahn	27-21-410-7100	S1	€120.00	25-Mar-2007
Arco Travertino	39-064885889	Via Valgia Silvilla 71, Roma	00179	John T. Okon	0181-123-5589	S1	€250.00	17-Jul-2007
Anne G. Farriss	0181-382-7185	2119 Elm St., Hampstead, London	NW3 RTA	Alex B. Alby	0161-228-1249	T2	€100.00	03-Dec-2007
Olette K. Smith	34-934412463	Avinguda del Paral.Lel 50, Barcelona	08001	John T. Okon	0181-123-5589	S2	€500.00	14-Mar-2007

C_NAME = Customer name
C_PHONE = Customer phone
C_ADDRESS = Customer address
C_POSTCODE = Customer postcode
A_NAME = Agent name
A_PHONE = Agent phone
TP = Insurance type
AMT = Insurance policy amount, in thousands of £
REN = Insurance renewal date

Historical Roots: Files and File Systems (continued)

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. The smallest piece of data that can be recognized by the computer is a single character, such as the letter *A*, the number 5, or a symbol such as /. A single character requires 1 byte of computer storage.
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 named J. D. Rudd might consist of J. D. Rudd'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 vendors of ROBCOR Company, or a file might contain the records for the students currently enrolled at Gigantic University.

Historical Roots: Files and File Systems (continued)

- DP specialist wrote programs for reports:
  - Monthly summaries of types and amounts of insurance sold by agents
  - Monthly reports about which customers should be contacted for renewal
  - Reports that analyzed ratios of insurance types sold by agent
  - Customer contact letters summarizing coverage

Historical Roots: Files and File Systems (continued)

- Other departments requested databases be written for them
  - SALES database created for sales department
  - AGENT database created for personnel department

Historical Roots: Files and File Systems (continued)

FIGURE 1.4 Contents of the AGENT file

A_NAME	A_PHONE	A_ADDRESS	POSTCODE	HIRED	YTD_PAY	YTD_IT	YTD_NI	YTD_SLS	DEP
Alex B. Alby	0161-228-1249	Dekon Van Erpstraat 20, Best	5492	01-Nov-1998	€20,806.00	€5,201.00	€1,684.00	€103,983.00	3
Leah F. Hahn	27-21-410-7100	West Quay Road, Waterfront, Cape Town	8002	23-May-1984	€25,230.00	€6,308.00	€2,018.00	€108,844.00	0
John T. Okon	0181-123-5589	452 Elm St., Hampstead, London	NW3 TYU	15-Jun-2003	€18,169.00	€4,542.00	€1,453.00	€99,548.00	2

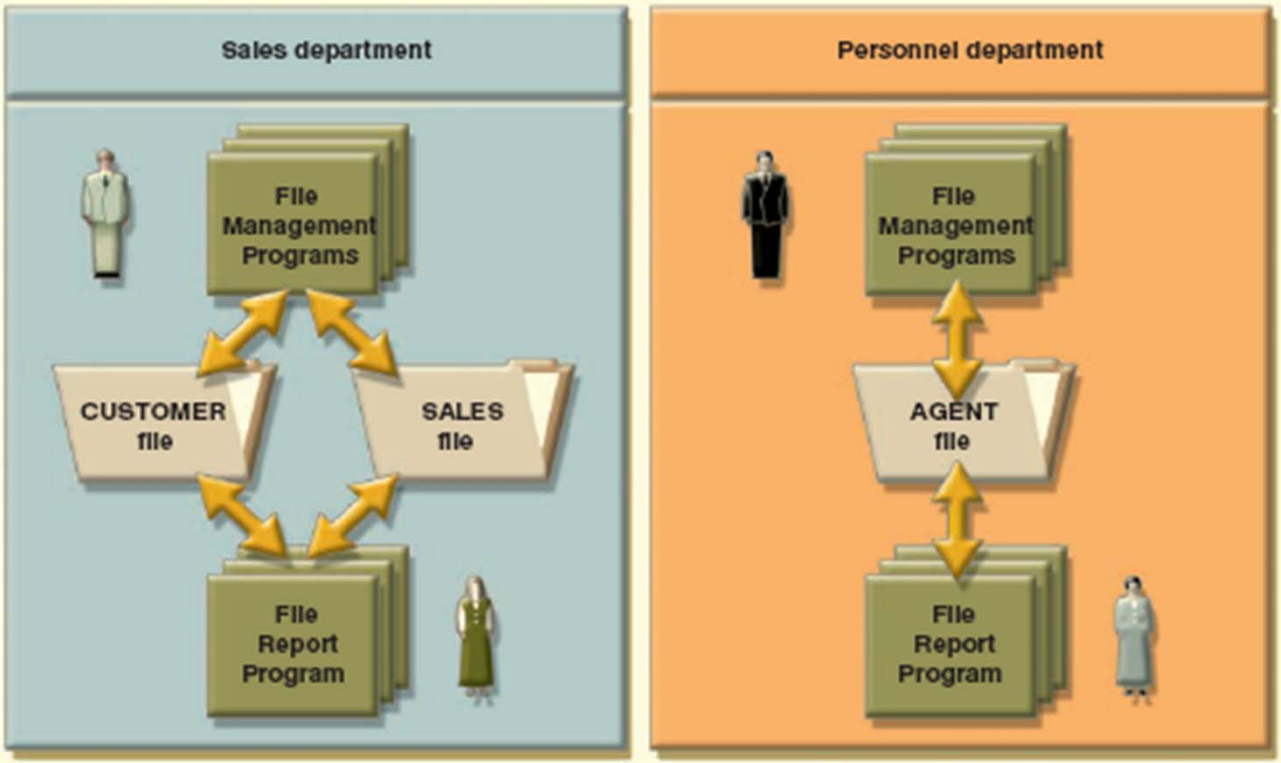
A_NAME = Agent name
A_PHONE = Agent phone
A_ADDRESS = Agent address
POSTCODE = Agent postcode
HIRED = Agent date of hire
YTD_PAY = Year-to-date pay
YTD_IT = Year-to-date income tax paid
YTD_NI = Year-to-date national insurance paid
YTD_SLS = Year-to-date sales
DEP = Number of dependants

Historical Roots: Files and File Systems (continued)

- As number of databases increased, small file system evolved
- Each file used its own application programs
- Each file was owned by individual or department who commissioned its creation

Historical Roots: Files and File Systems (continued)

FIGURE 1.5 A simple file system



Example of Early Database Design (continued)

- As system grew, demand for DP's programming skills grew
- Additional programmers hired
- DP specialist evolved into DP manager, supervising a DP department
- Primary activity of department (and DP manager) remained programming

Problems with File System Data Management

- Every task requires extensive programming in a third-generation language (3GL)
  - Programmer must specify task and how it must be done
- Modern databases use fourth-generation languages (4GL)
  - Allow users to specify what must be done without specifying how it is to be done

Problems with File System Data Management

TABLE 1.3 3GL vs. 4GL sample code

3GL (Generic Code)	4GL (SQL Code)
DO WHILE NOT EOF() READ CUSTOMER IF CUSTOMER.C_POSTCODE = 'M15 BVC' THEN PRINT C_NAME, C_PHONE, C_ POSTCODE; ENDDO;	SELECT C_NAME, C_PHONE, C_POSTCODE FROM CUSTOMER WHERE CUSTOMER.C_POSTCODE = ' M15 BVC';

Problems with File System Data Management

- Time-consuming, high-level activity
- As number of files expands, system administration becomes difficult
- Making changes in existing file structure is difficult
- File structure changes require modifications in all programs that use data in that file

Problems with File System Data Management

- Modifications are likely to produce errors, requiring additional time to "debug" the program
- Security features hard to program and therefore often omitted

Structural and Data Dependence

- Structural dependence
  - Access to a file depends on its structure

Structural and Data Dependence (continued)

- Data dependence
  - Changes in the data storage characteristics without affecting the application program's ability to access the data
  - Logical data format
    - How the human being views the data
  - Physical data format
    - How the computer "sees" the data

Field Definitions and Naming Conventions

- Flexible record definition anticipates reporting requirements by breaking up fields into their component parts

Field Definitions and Naming Conventions (continued)

TABLE 1.4 Sample customer file fields

Field	Contents	Sample entry
CUS_LNAME	Customer last name	Ramas
CUS_FNAME	Customer first name	Alfred
CUS_INITIAL	Customer initial	A
CUS_AREACODE	Customer area code	1615
CUS_PHONE	Customer phone	0161-234-5678
CUS_ADDRESS	Customer street address or box number	123 Green Meadow Lane
CUS_CITY	Customer city	Murfreesboro
CUS_COUNTY	Customer county/district	Lancashire
CUS_POSTCODE	Customer postcode	M14 TYR

Data Redundancy

- Data redundancy results in data inconsistency
  - Different and conflicting versions of the same data appear in different places
- Errors more likely to occur when complex entries are made in several different files and/or recur frequently in one or more files
- Data anomalies develop when required changes in redundant data are not made successfully

Data Redundancy

Types of data anomalies:

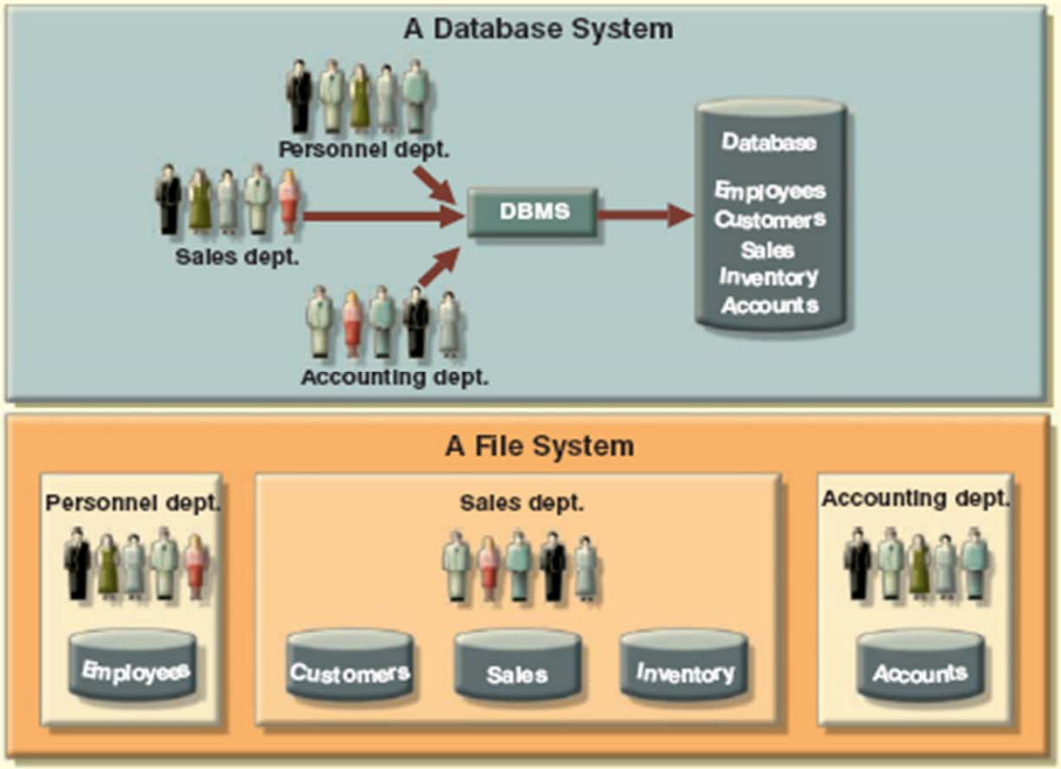
- Update anomalies
  - Occur when changes must be made to existing records
- Insertion anomalies
  - Occur when entering new records
- Deletion anomalies
  - Occur when deleting records

Database Systems

- Problems inherent in file systems make using a database system desirable
- File system
  - Many separate and unrelated files
- Database
  - Logically related data stored in a single logical data repository

Database Systems

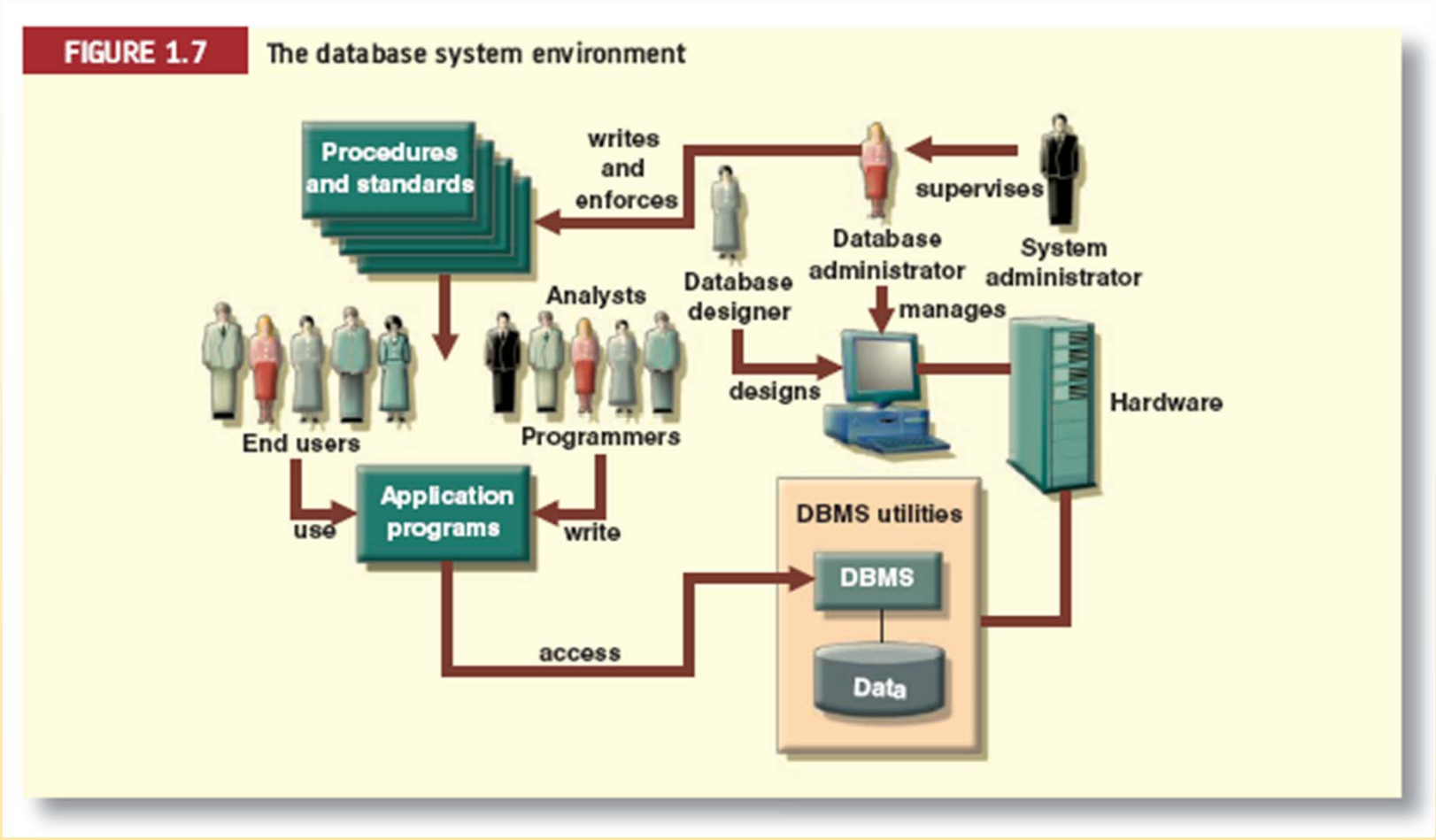
FIGURE 1.6 Contrasting database and file systems



The Database System Environment

- Database system is composed of five main parts:
  - Hardware
  - Software
    - Operating system software
    - DBMS software
    - Application programs and utility software
  - People
  - Procedures
  - Data

The Database System Environment (continued)



DBMS Functions

- DBMS performs functions that guarantee integrity and consistency of data
  - Data dictionary management
    - defines data elements and their relationships
  - Data storage management
    - stores data and related data entry forms, report definitions, etc.

DBMS Functions (continued)

- Data transformation and presentation
  - translates logical requests into commands to physically locate and retrieve the requested data
- Security management
  - enforces user security and data privacy within database

DBMS Functions (continued)

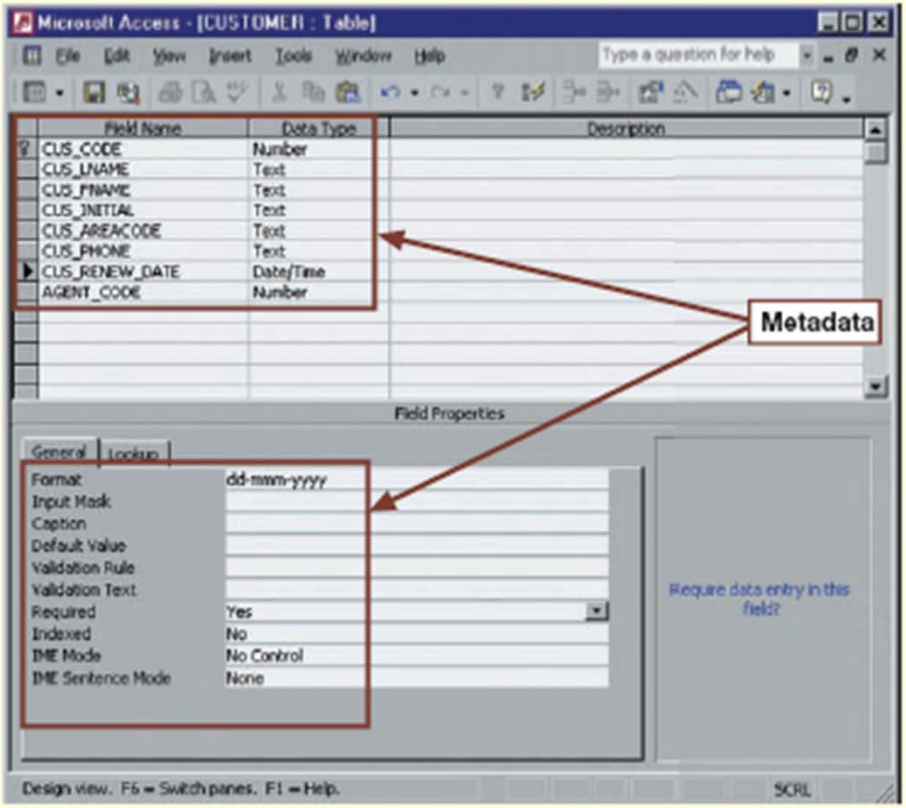
- Multiuser access control
  - uses sophisticated algorithms to ensure multiple users can access the database concurrently without compromising the integrity of the database
- Backup and recovery management
  - provides backup and data recovery procedures
- Data integrity management
  - promotes and enforces integrity rules

DBMS Functions (continued)

- Database access languages and application programming interfaces
  - provide data access through a query language
- Database communication interfaces
  - allow database to accept end-user requests via multiple, different network environments

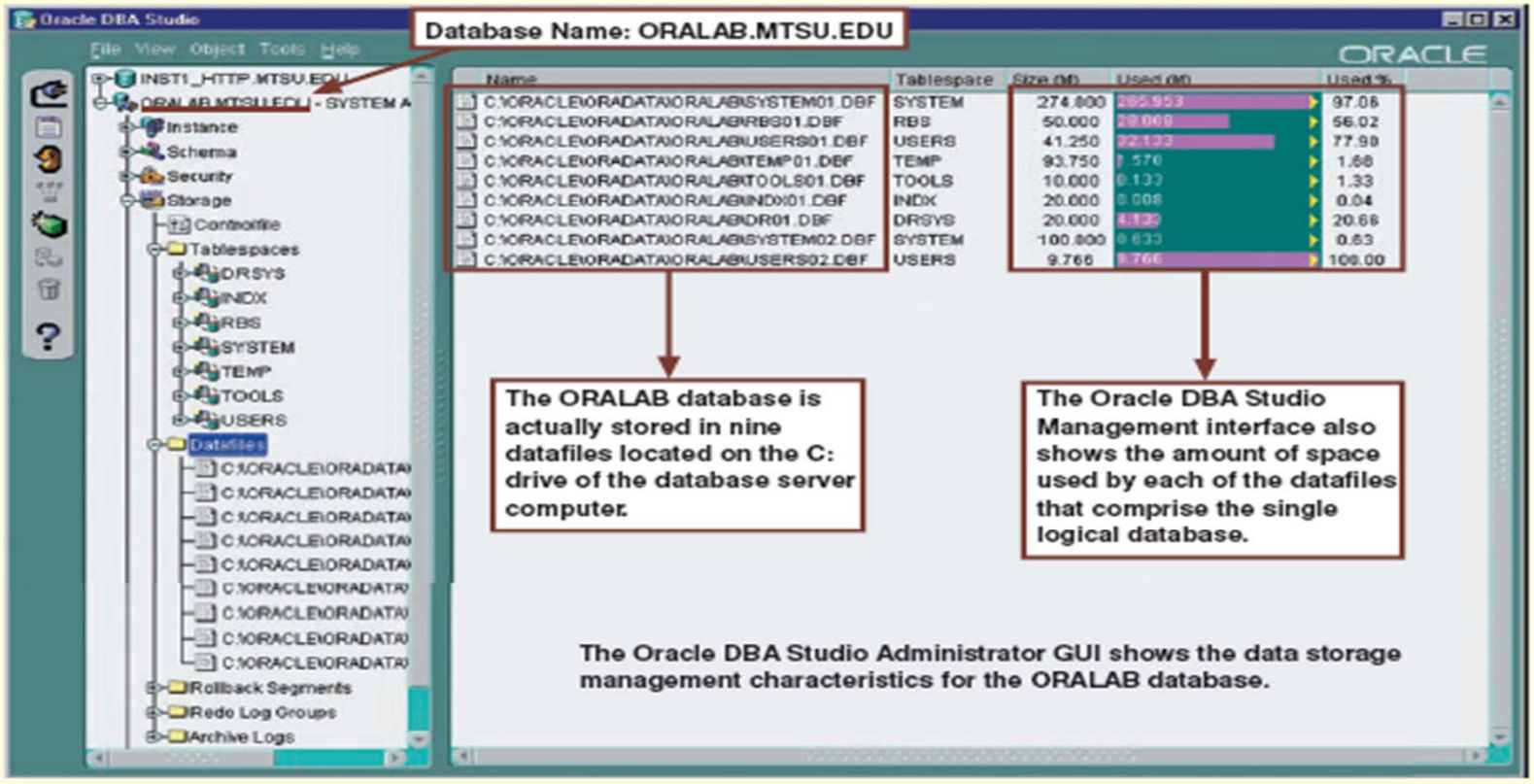
DBMS Functions (continued)

FIGURE 1.8 Illustrating metadata with Microsoft Access



DBMS Functions (continued)

FIGURE 1.9 Illustrating data storage management with Oracle



Summary

- Data are raw facts. Information is the result of processing data to reveal its meaning.
- To implement and manage a database, use a DBMS.
- Database design defines the database structure.
- A well-designed database facilitates data management and generates accurate and valuable information.
- A poorly designed database can lead to bad decision making, and bad decision making can lead to the failure of an organization.

Summary (continued)

- Databases were preceded by file systems.
- Limitations of file system data management:
  - requires extensive programming
  - system administration complex and difficult
  - making changes to existing structures is difficult
  - security features are likely to be inadequate
  - independent files tend to contain redundant data
- DBMS's were developed to address file systems' inherent weaknesses