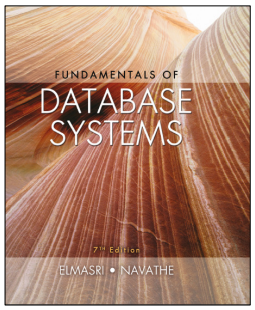
Fundamentals of Database Systems

Seventh Edition



Chapter 1

Databases and Database Users

Pearson

1

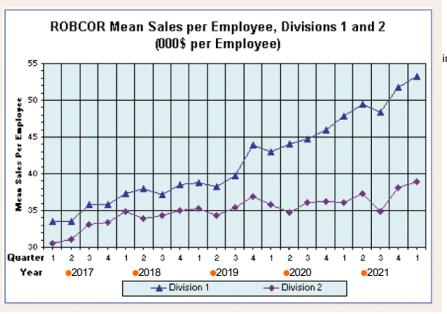
Introducing the Database

- Data
  - raw facts
    - "raw" indicates that the facts have not been *processed* to reveal their meaning
  - Data are the foundation of information, which is the bedrock of knowledge
- Information
  - processed data
    - on which to base decisions
  - Information requires context to reveal meaning

Pearson

3

ROBCOR Company



invoice number = 300124
invoice date = 12- JUN-2020
sales amount = $ 125.98

Sales productivity per Employee for each Division?

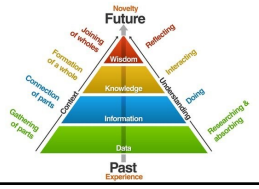
SALES PER EMPLOYEE FOR ROBCOR'S TWO DIVISIONS

Pearson

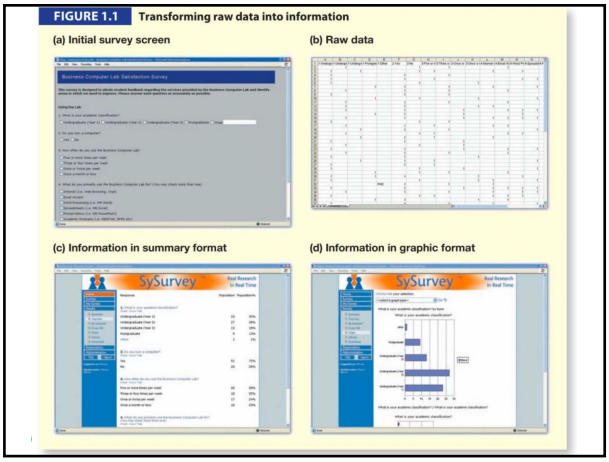
4

Introducing the Database (Con't.)

- Data versus Information
  - Data: building blocks of information
  - Information is produced by processing data
  - Information used to reveal meaning in data
  - Good, timely, relevant information is the key to good decision making
  - Good decision making is the key to organizational survival



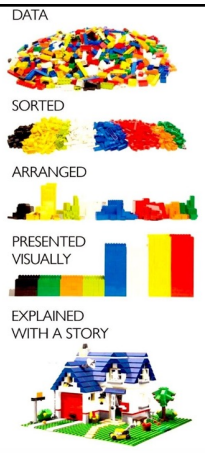
5



6

Database Management

- Good data ⟶ Good information
- Data must be generated, stored, and retrieved properly
- The data environment must be managed carefully
- Data Management is a discipline that focuses on the proper generation, storage, and retrieval of data

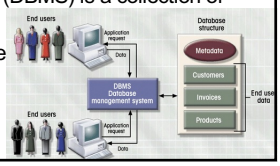


Pearson

7

Database Management (Con't.)

- 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.
- Database Management System (DBMS) is a collection of programs that,
  - Manages Database structure
  - Controls access to data
  - Contains query language



Pearson

9

Importance of DBMS

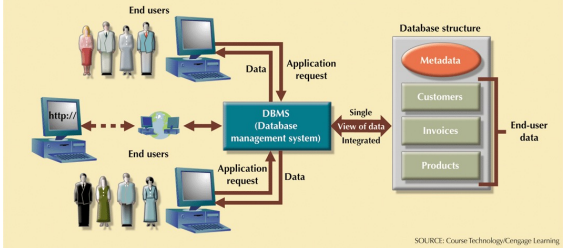
- Makes data management more efficient and effective
- Contains a *query language* that allows quick answers to ad hoc queries
  - Query is a question.
  - Ad hoc query is a spur-of-the-moment question.
- Provides better access to more and better-managed data
- Promotes integrated view of organization's operations
- Reduces the probability of inconsistent data

Pearson

10

DBMS Manages Interaction

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



DBMS is a collection of programs that manages the database structure and controls access to the data stored in the database.

Pearson

11

Basic Definitions

- **Database:**
  - A collection of related data.
- **Data:**
  - Known facts that can be recorded and have an implicit meaning.
- **Mini-world:**
  - Some part of the real world about which data is stored in a database. For example, student grades and transcripts at a university.
- **Database Management System (DBMS):**
  - A software package/ system to facilitate the creation and maintenance of a computerized database.
- **Database System:**
  - The DBMS software together with the data itself. Sometimes, the applications are also included.

Pearson

12

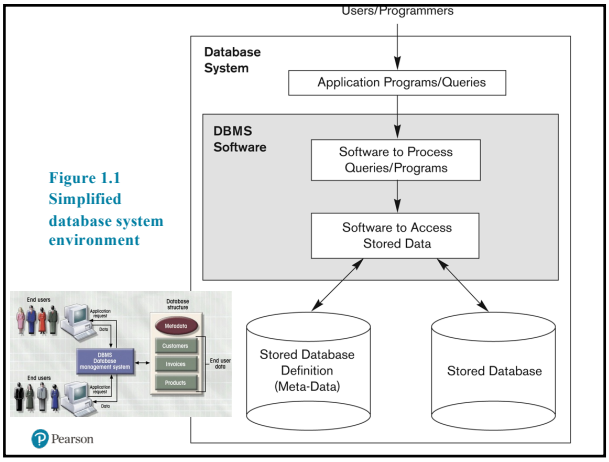


Figure 1.1 Simplified database system environment

Pearson

13

Types of Databases and Database Applications

- Traditional Applications:
  - Numeric and Textual Databases
- More Recent Applications:
  - Multimedia Databases
  - Geographic Information Systems (GIS)
  - Biological and Genome Databases
  - Data Warehouses
  - Mobile databases
  - Real-time and Active Databases

Pearson

14

Recent Developments (1 of 2)

- Social Networks started capturing a lot of information about people and about communications among people- posts, tweets, photos, videos in systems such as:
  - Facebook, Twitter, Linked-In
  - All of the above constitutes data
- NOSQL (Not Only SQL) systems
  - NOSQL (Not Only SQL- where SQL is the de facto standard language for relational DBMSs) systems have been designed for rapid search and retrieval from documents, processing of huge graphs occurring on social networks, and other forms of unstructured data with flexible models of transaction processing
- A large amount of data now resides on the "cloud" which means it is in huge data centers using thousands of machines.

Pearson

15

Recent Developments (2 of 2)

- Search Engines- Google, Bing, Yahoo : collect their own repository of web pages for searching purposes
- New Technologies are emerging from the so-called non-database software vendors to manage vast amounts of data generated on the web:
- Big Data storage systems involving large clusters of distributed computers
  - New data storage, management and analysis technology was necessary to deal with the onslaught of data in petabytes a day (10^{15} bytes or 1000 terabytes) in some applications – this started being commonly called as "Big Data".
  - Hadoop (which originated from Yahoo) and Mapreduce Programming approach to distributed data processing (which originated from Google) as well as the Google file system have given rise to Big Data technologies. Further enhancements are taking place in the form of Spark based technology

Pearson

16

Impact of Databases and Database Technology

- **Businesses:** Banking, Insurance, Retail, Transportation, Healthcare, Manufacturing
- **Service Industries:** Financial, Real-estate, Legal, Electronic Commerce, Small businesses
- **Education:** Resources for content and Delivery
- **More recently:** Social Networks, Environmental and Scientific Applications, Medicine and Genetics
- **Personalized Applications:** based on smart mobile devices

Pearson

17

What are databases?

- Holds data
- Organizes data
- Retrieve/Search data through DBMS

A usually large collection of data organized especially for rapid search and retrieval.

Databases and file storage

Databases

- Very organized
- Functionality like search, replication, ...

File systems

- Less organized
- Simple, less added functionality

Pearson

18

Structured and unstructured data

Structured: database schema

- Relational database

Semi-structured

```
{ "key": "value"}
```

- JSON (JavaScript Object Notation)

Unstructured: schemaless, more like files

- Videos, photos

SQL and NoSQL

SQL
- Tables
- Database schema
- Relational databases

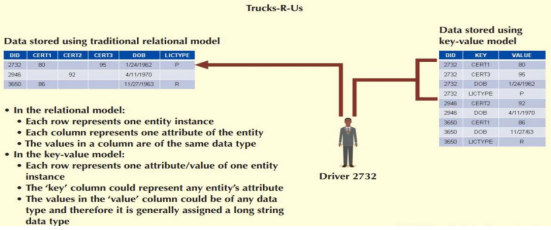
NoSQL
- Non-relational databases
- Structured or unstructured
- Key-value stores (e.g. caching)
- Document DB (e.g. JSON objects)

MySQL PostgreSQL redis mongoDB.

Pearson

19

FIGURE 2.6 A simple key-value representation



SQL
- Tables
- Database schema
- Relational databases

NoSQL
- Non-relational databases
- Structured or unstructured
- Key-value stores (e.g. caching)
- Document DB (e.g. JSON objects)

MySQL PostgreSQL redis mongoDB.

Pearson

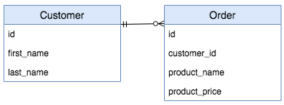
20

A schema describes the structure and relations of a database

SQL: The database schema

```
-- Create Customer Table
CREATE TABLE "Customer" (
  "id" SERIAL NOT NULL,
  "first_name" varchar,
  "last_name" varchar,
  PRIMARY KEY ("id")
);

-- Create Order Table
CREATE TABLE "Order" (
  "id" SERIAL NOT NULL,
  "customer_id" integer REFERENCES "Customer",
  "product_name" varchar,
  "product_price" integer,
  PRIMARY KEY ("id")
);
```




```
-- Join both tables on foreign key
SELECT * FROM "Customer"
INNER JOIN "Order"
ON "customer_id" = "Customer"."id";
```

```
id | first_name | ... | product_price
 1 | Vincent    | ... |            10
```

Pearson

23

Typical DBMS Functionality

- **Define** a particular database in terms of its data types, structures, and constraints
- **Construct** or Load the initial database contents on a secondary storage medium
- **Manipulating** the database:
  - Retrieval: Querying, generating reports
  - Modification: Insertions, deletions and updates to its content
  - Accessing the database through Web applications
- **Processing** and **Sharing** by a set of concurrent users and application programs – yet, keeping all data valid and consistent

Pearson

25

Why Database Design is Important

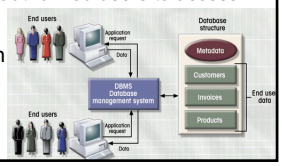
- Database design focuses on design of database structure used for end-user data
  - Designer must identify database's expected use
- Well-designed database:
  - Facilitates data management
  - Generates accurate and valuable information
- Poorly designed database:
  - Causes difficult-to-trace errors
  - Poor design results in unwanted data redundancy
    - Redundant data is unnecessarily duplicated data.
  - Poor design generates errors leading to bad decisions

Pearson

26

Application Activities Against a Database

- Applications interact with a database by generating
  - **Queries:** that access different parts of data and formulate the result of a request
  - **Transactions:** that may read some data and "update" certain values or generate new data and store that in the database
- Applications must not allow unauthorized users to access data
- Applications must keep up with changing user requirements against the database



Pearson

27

Example of a Database (with a Conceptual Data Model) (1 of 2)

- **Mini-world for the example:**
  - Part of a UNIVERSITY environment.
- **Some mini-world entities:**
  - STUDENTs
  - COURSEs
  - SECTIONs (of COURSEs)
  - (academic) DEPARTMENTs
  - INSTRUCTORs

Pearson

29

Example of a Database (with a Conceptual Data Model) (2 of 2)

- **Some mini-world relationships:**
  - SECTIONs **are of specific** COURSEs
  - STUDENTs **take** SECTIONs
  - COURSEs **have prerequisite** COURSEs
  - INSTRUCTORs **teach** SECTIONs
  - COURSEs **are offered by** DEPARTMENTs
  - STUDENTs **major in** DEPARTMENTs
- Note: The above entities and relationships are typically expressed in a conceptual data model, such as the ENTITY-RELATIONSHIP data model

Pearson

30

Figure 1.2 Example of a simple database (1 of 4)

COURSE

Course_name	Course_number	Credit_hours	Department
Intro to Computer Science	CS1310	4	CS
Data Structures	CS3320	4	CS
Discrete Mathematics	MATH2410	3	MATH
Database	CS3380	3	CS

Pearson

31

Figure 1.2 Example of a simple database (2 of 4)

SECTION

Section_identifier	Course_number	Semester	Year	Instructor
85	MATH2410	Fall	07	King
92	CS1310	Fall	07	Anderson
102	CS3320	Spring	08	Knuth
112	MATH2410	Fall	08	Chang
119	CS1310	Fall	08	Anderson
135	CS3380	Fall	08	Stone

Pearson

32

Figure 1.2 Example of a simple database (3 of 4)

GRADE_REPORT

Student_number	Section_identifier	Grade
17	112	B
17	119	C
8	85	A
8	92	A
8	102	B
8	135	A

Pearson

33

Figure 1.2 Example of a simple database (4 of 4)

PREREQUISITE

Course_number	Prerequisite_number
CS3380	CS3320
CS3380	MATH2410
CS3320	CS1310

Pearson

34

Main Characteristics of the Database Approach (1 of 3)

- **Self-describing nature of a database system:**
  - A DBMS **catalog** stores the description of a particular database (e.g. data structures, types, and constraints)
  - The description is called **meta-data***.
  - This allows the DBMS software to work with different database applications.
- **Insulation between programs and data:**
  - Called **program-data independence**.
  - Allows changing data structures and storage organization without having to change the DBMS access programs.

* Some newer systems such as a few NOSQL systems need no meta-data: they store the data definition within its structure making it self describing

Pearson

35

Figure 1.3 Example of a simplified database catalog (1 of 2)

RELATIONS (Tables)

Relation_name	No_of_columns
STUDENT	4
COURSE	4
SECTION	5
GRADE_REPORT	3
PREREQUISITE	2

Pearson

36

Figure 1.3 Example of a simplified database catalog (2 of 2)

COLUMNS

Column_name	Data_type	Belongs_to_relation
Name	Character (30)	STUDENT
Student_number	Character (4)	STUDENT
Class	Integer (1)	STUDENT
Major	Major_type	STUDENT
Course_name	Character (10)	COURSE
Course_number	XXXXNNNN	COURSE
....	....	....
....	....	....
....	....	....
Prerequisite_number	XXXXNNNN	PREREQUISITE

Note: Major_type is defined as an enumerated type with all known majors. XXXXNNNN is used to define a type with four alphabetic characters followed by four numeric digits.

Pearson

37

Main Characteristics of the Database Approach (2 of 3)

- **Data Abstraction:**
  - A **data model** is used to hide storage details and present the users with a conceptual view of the database.
  - Programs refer to the data model constructs rather than data storage details
- **Support of multiple views of the data:**
  - Each user may see a different view of the database, which describes **only** the data of interest to that user.

Pearson

38

Main Characteristics of the Database Approach (3 of 3)

- **Sharing of data and multi-user transaction processing:**
  - Allowing a set of **concurrent users** to retrieve from and to update the database.
  - **Concurrency control** within the DBMS guarantees that each **transaction** is correctly executed or aborted
  - **Recovery** subsystem ensures each completed transaction has its effect permanently recorded in the database
  - **OLTP** (Online Transaction Processing) is a major part of database applications. This allows hundreds of concurrent transactions to execute per second.

Pearson

39

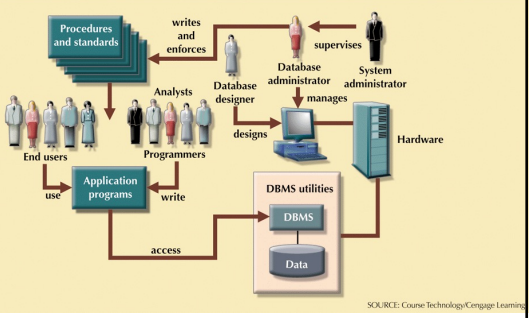
Database Users

- Users may be divided into
  - Those who actually use and control the database content, and those who design, develop and maintain database applications (called "Actors on the Scene"), and
  - Those who design and develop the DBMS software and related tools, and the computer systems operators (called "Workers Behind the Scene").

Pearson

40

FIGURE 1.9 The database system environment



Database system refers to an organization of components that define and regulate the collection, storage, management, and the use of data within a database environment.

Pearson

41

Database Users – Actors on the Scene (1 of 2)

- Actors on the scene
  - **Database administrators:**
    - Responsible for authorizing access to the database, for coordinating and monitoring its use, acquiring software and hardware resources, controlling its use and monitoring efficiency of operations.
  - **Database Designers:**
    - Responsible to define the content, the structure, the constraints, and functions or transactions against the database. They must communicate with the end-users and understand their needs.
  - **End-users:** They use the data for queries, reports and some of them update the database content.

Pearson

42

Database Users – Actors on the Scene (2 of 2)

- **System Analysts and Application Developers**

  This category currently accounts for a very large proportion of the IT work force.

  - **System Analysts**: They understand the user requirements of naïve and sophisticated users and design applications including canned transactions to meet those requirements.
  - **Application Programmers:** Implement the specifications developed by analysts and test and debug them before deployment.
  - **Business Analysts**: There is an increasing need for such people who can analyze vast amounts of business data and real-time data ("Big Data") for better decision making related to planning, advertising, marketing etc.

Pearson

43

Database System Environment (Con't.)

– Hardware - the system's physical devices

– computer, its peripherals (keyboard, mouse, modems, printers), electronic devices (network)

- Software - the collection of programs

– Operating systems software (Windows, UNIX, etc.)

– DBMS software (MS Access, SQL Server, Oracle, MySQL, PostgreSQL etc.)

– Application programs and utility software

- People - all users of the database system

– System administrators, oversee the system's general operations

– Database administrators, manage the DBMS's use, ensure that the database is functioning properly

– Database designers, design the database structure

– System analysts and programmers, design & implement appl. prog.

– End users, use the application programs to run the daily operations

Pearson

44

Database System Environment (Con't.)

- Procedures

– the instructions and rules that govern the design and use of the database system

– ensure that there is an organized way to monitor and audit both the data that enter the database and the information that is generated through the use of such data

- Data

– the collection of facts

NOTE that

- The existence of a database system adds a new dimension to an organization's management structure.
- The managerial structure depends on the organization's size and its functions.
- 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

Pearson

45

Historical Roots of Database

- First applications focused on office tasks: order/entry processing, financial salary sheets, work scheduling, etc.
- Requests for information quickly followed
- File systems developed to address needs
  - Data organized according to expected use
  - Data Processing (DP) specialists computerized manual file systems

Pearson

54

FIGURE 1.5 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., Babs, TN	36123	Leah F. Hahn	615-882-1244	T1	100.00	05-Apr-2012
Leona K. Dunne	713-894-1238	Box 12A, Fox, KY	25246	Alex B. Alby	713-228-1249	T1	250.00	16-Jun-2012
Kathy W. Smith	615-894-2285	125 Oak Ln, Babs, TN	36123	Leah F. Hahn	615-882-2144	S2	150.00	29-Jan-2013
Paul F. Olowski	615-894-2180	217 Lee Ln., Babs, TN	36123	Leah F. Hahn	615-882-1244	S1	300.00	14-Oct-2012
Myron Orlando	615-222-1672	Box 111, New, TN	36155	Alex B. Alby	713-228-1249	T1	100.00	28-Dec-2012
Amy B. O'Brian	713-442-3381	387 Troll Dr., Fox, KY	25246	John T. Okon	615-123-5589	T2	850.00	22-Sep-2012
James G. Brown	615-297-1228	21 Tye Rd., Nash, TN	37118	Leah F. Hahn	615-882-1244	S1	120.00	25-Mar-2013
George Williams	615-290-2556	155 Maple, Nash, TN	37119	John T. Okon	615-123-5589	S1	250.00	17-Jul-2012
Anne G. Farriss	713-382-7185	2119 Elm, Crew, KY	25432	Alex B. Alby	713-228-1249	T2	100.00	03-Dec-2012
Olette K. Smith	615-297-3809	2782 Main, Nash, TN	37118	John T. Okon	615-123-5589	S2	500.00	14-Mar-2013

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

SOURCE: Course Technology/Cengage

Pearson

55

File Terminology

- Data
  - Raw Facts
- Field
  - Group of characters with specific meaning
- Record
  - Logically connected fields that describe a person, place, or thing
- File
  - Collection of related records

Pearson

56

A_NAME	A_PHONE	A_ADDRESS	ZIP	HIRED	YTD_PAY	YTD_FIT	YTD_FICA	YTD_SLS	DEP
Alex B. Alby	713-228-1249	123 Toll, Nash, TN	37119	11.01.1993	$20.566,24	$4.332,21	$1.534,57	$1.735,00	3
Leah F. Hahn	615-882-1244	334 Main, Fox, KY	25246	23.05.1984	$25.213,76	$5.934,75	$1.788,36	$4.967,00	0
John T. Okon	615-123-5589	452 Elm, New, TN	36155	15.06.1989	$23.198,29	$4.332,24	$1.689,44	$3.093,00	2

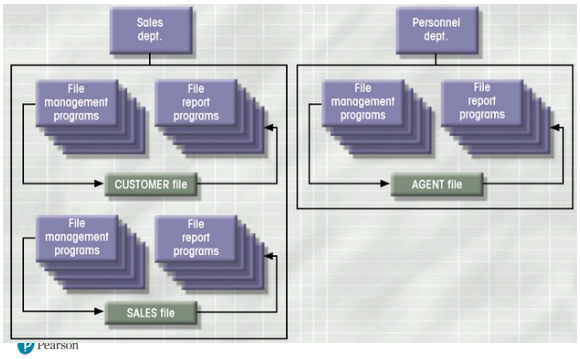
A_NAME = Agent name
A_PHONE = Agent phone
A_ADDRESS = Agent address
ZIP = Agent ZIP code
HIRED = Agent date of hire
YTD_PAY = Year-to-date pay
YTD_FIT = Year-to-date federal income tax paid
YTD_FICA = Year-to-date Social Security paid
YTD_SLS = Year-to-date sales, in thousands of $
DEP = Number of dependents

CONTENTS OF THE AGENT FILE

Pearson

57

Simple File System

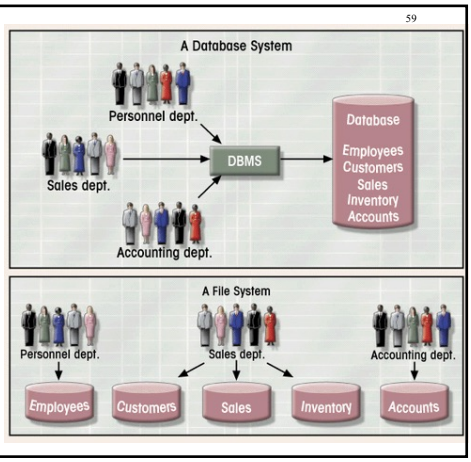


Pearson

58

59

Database vs. File Systems



Pearson

59

File System Critique

- Understanding the shortcomings of the file system enables us to understand the reasons for the development of the database
- Many of the problems are not unique to the file systems
- File System Data Management
  - Requires extensive programming
  - Time consuming
  - Makes ad hoc queries impossible
  - Lack of security
  - Leads to islands of information (different versions of the same data)

Pearson

60

Structural and Data Dependence

- Structural Dependence (access to a file is dependent on its structure)
  - All file system programs must be modified to conform to a new file structure
- Data Dependence
  - Data access changes when data storage characteristics change (Change in file's data characteristics requires modification of data access programs)
  - Must tell program what to do and how to do it
  - Makes file systems cumbersome from programming and data management views

Pearson

61

62

Database Systems

- Database consists of logically related data stored in a single repository (Unlike the file system, with its many separate and unrelated files)
- Provides advantages over file system management approach
  - Eliminates data inconsistency, data anomalies, data dependency, and structural dependency problems
  - Stores not only the data structures, but also the relationships between those structures, and access paths to those structures

Pearson

62

Field Definitions & Naming Convensions

- Flexible field definition suitable for reporting requirements
- Selection of proper field names important (indicating the field's origin & descriptive of its content)
- Attention to length of field names
- Use of unique record identifiers

Pearson

63

Field Definitions & Naming Convensions

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	615
CUS_PHONE	Customer phone	844-2573
CUS_ADDRESS	Customer street address or box number	218 Fork Rd.
CUS_CITY	Customer city	Babs
CUS_STATE	Customer state	TN
CUS_ZIP	Customer zip code	36123

Pearson

64

Data Redundancy

- Same data stored unnecessarily in different places
- Results of uncontrolled data redundancy
  - Data inconsistency
    - Different and conflicting versions of same data (Lack of data integrity)
  - Data anomalies
    - Modification
    - Insertion
    - Deletion

Pearson

65

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
  - Vital in the data design process
- Good data modeling facilitates communication between the designer, user, and the developer

Pearson

66

Historical Development of Database Technology (1 of 3)

- **Early Database Applications:**
  - The Hierarchical and Network Models were introduced in mid 1960s and dominated during the seventies.
  - A bulk of the worldwide database processing still occurs using these models, particularly, the hierarchical model using IBM's IMS system.
- **Relational Model based Systems:**
  - Relational model was originally introduced in 1970, was heavily researched and experimented within IBM Research and several universities.
  - Relational DBMS Products emerged in the early 1980s.

Pearson

67

Historical Development of Database Technology (2 of 3)

- **Object-oriented and emerging applications:**
  - Object-Oriented Database Management Systems (OODBMSs) were introduced in late 1980s and early 1990s to cater to the need of complex data processing in CAD and other applications.
    - Their use has not taken off much.
  - Many relational DBMSs have incorporated object database concepts, leading to a new category called **object-relational** DBMSs (ORDBMSs)
  - **Extended relational** systems add further capabilities (e.g. for multimedia data, text, XML, and other data types)

Pearson

68

Historical Development of Database Technology (3 of 3)

- **Data on the Web and E-commerce Applications:**
  - Web contains data in HTML (Hypertext markup language) with links among pages.
  - This has given rise to a new set of applications and E-commerce is using new standards like XML (eXtended Markup Language).
  - Script programming languages such as PHP and JavaScript allow generation of dynamic Web pages that are partially generated from a database
    - Also allow database updates through Web pages