UNIT 4

MANAGING DATA RESOURCES

Lesson Structure

Contents

4.Objective

After going through this unit you should be able to

- Know about the organization of data in a file environment
- Understand the problem related to traditional file environment
- Learn the Database Management System and its different types
- learn about the data administrator, data warehouse, data mining.

4.1 Introduction

This chapter examines the managerial and organizational requirements as well as the technologies for managing data as a resource. Organizations need to manage their data assets very carefully to make sure that the data can be easily accessed and used by managers and employees across the organization. We describe the typical challenges facing businesses trying to access information using traditional file management technologies. Then we describe the technology of database management systems, which can overcome many of the drawbacks of traditional file management and provide the firm wide integration of information required for digital firm applications. We include a discussion of the managerial and organizational requirements for successfully implementing a database environment.

4.2 Organizing Data in a Traditional File Environment

An effective information system provides users with timely, accurate, and relevant information. This information is stored in computer files. When the files are properly arranged and maintained, users can easily access and retrieve the information they need. Well-managed, carefully arranged files make it easy to obtain data for business decisions, whereas poorly managed files lead to chaos in information processing, high costs, poor performance, and little, if any, flexibility. Despite the use of excellent hardware and software, many organizations have inefficient information systems because of poor file management. In this section we describe the traditional methods that organizations have used to arrange data in computer files. We also discuss the problems with these methods.

4.3 File Organization Terms and concepts

A computer system organizes data in a hierarchy that starts with bits and bytes and progresses to fields, records, files, and databases. A bit represents the smallest unit of data a computer can handle. A group of bits, called a byte, represents a single character, which can be a letter, a number, or another symbol. A grouping of characters into a word, a group of words, or a complete number (such as a person's name or age) is called a field. A group of related fields, such as the student's name, the course taken, the date, and the grade, comprises a record; a group of records of the same type is called a file. For instance, the student records in Figure could constitute a course file. A group of related files makes up a database. The student course file illustrated in Figure 4-1 could be grouped with files on students' personal histories and financial backgrounds to create a student database.

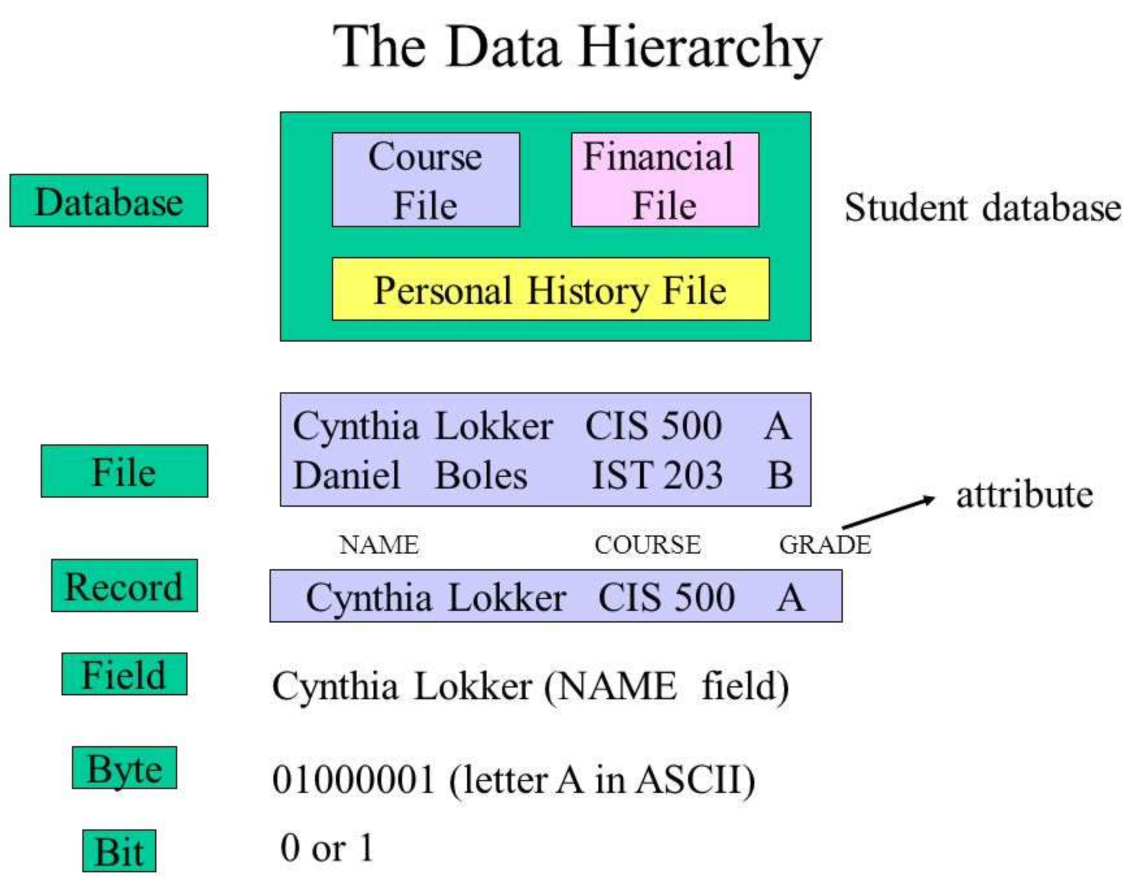

Figure 4-1

A computer system organizes data in a hierarchy that starts with the bit, which represents either a 0 or a 1. Bits can be grouped to form a byte to represent one character, number, or symbol. Bytes can be grouped to form a field, and related fields can be grouped to form a record. Related records can be collected to form a file, and related files can be organized into a database.

A record describes an entity. An entity is a person, place, thing, or event on which we maintain information. An order is a typical entity in a sales order file, which maintains information on a firm's sales orders. Each characteristic or quality describing a particular entity is called an attribute. For example, order number, order date, order amount, item number, and item quantity would each be an attribute of the entity order. The specific values that these attributes can have can be found in the fields of the record describing the entity order (see Figure 4-2).

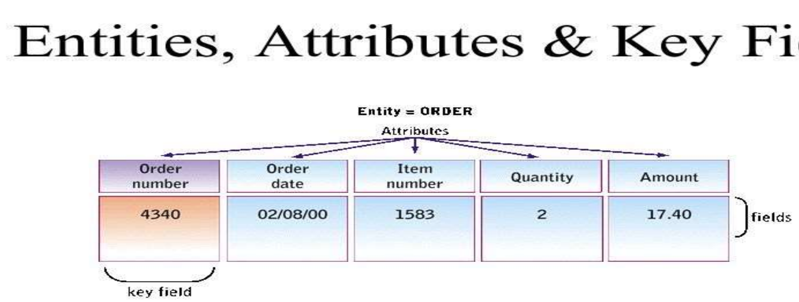

Figure 4-2

Every record in a file should contain at least one field that uniquely identifies that record so that the record can be retrieved, updated, or sorted. This identifier field is called a key field. An example of a key field is the order number for the order record illustrated in Figure 4-2 or an employee number or social security number for a personnel record (containing employee data such as the employee's name, age, address, job title, and so forth).

4.4 Problems with the Traditional File Environment

In most organizations, systems tended to grow independently, and not according to some grand plan. Each functional area tended to develop systems in isolation from other functional areas.

Accounting, finance, manufacturing, human resources, sales, and marketing all developed their own systems and data files. Each application, of course, required its own files and its own computer program to operate. For example, the human resources functional area might have a personnel master file, a payroll file, a medical insurance file, a pension file, a mailing list file, and so forth until tens, perhaps hundreds, of files and programs existed. In the company as a whole, this process led to multiple master files created, maintained, and operated by separate divisions or departments. As this process goes on for five or ten years, the organization is saddled with hundreds of programs and applications, with no one who knows what they do, what data they use, and who is using the data. The resulting problems are data redundancy, program-data dependence, inflexibility, poor data security, and inability to share data among applications.

4.4.1 Data Redundancy and Confusion

Data redundancy is the presence of duplicate data in multiple data files. Data redundancy occurs when different divisions, functional areas, and groups in an organization independently collect the same piece of information. For instance, within the commercial loans division of a bank, the marketing and credit information functions might collect the same customer information. Because it is collected and maintained in so many different places, the same data item may have different meanings in different parts of the organization. Simple data items, such as the fiscal year, employee identification, and product code, can take on different meanings as programmers and analysts work in isolation on different applications.

4.4.2Program-Data Dependence

Program-data dependence is the tight relationship between data stored in files and the specific programs required to update and maintain those files. Every computer program has to describe the location and nature of the data with which it works. In a traditional file environment, any change in data requires a change in all programs that access the data. Changes, for instance, in tax rates or ZIP code length require changes in programs. Such programming changes may cost millions of dollars to implement in programs that require the revised data.

4.4.3Lack of Flexibility

A traditional file system can deliver routine scheduled reports after extensive programming efforts, but it cannot deliver ad hoc reports or respond to unanticipated information requirements in a timely fashion. The information required by ad hoc requests is somewhere in the system but too expensive to retrieve. Several programmers would have to work for weeks to put together the required data items in a new file.

4.4.4Poor Security
Because there is little control or management of data, access to and dissemination of information may be out of control. Management may have no way of knowing who is accessing or even making changes to the organization's data.

4.4.5Lack of Data-Sharing and Availability
The lack of control over access to data in this confused environment does not make it easy for people to obtain information. Because pieces of information in different files and different parts of the organization cannot be related to one another, it is virtually impossible for information to be shared or accessed in a timely manner. Information cannot flow freely across different functional areas or different parts of the organization.

4.5 The Database Approach to Data Management
Database technology can cut through many of the problems a traditional file organization creates. A more rigorous definition of a database is a collection of data organized to serve many applications efficiently by centralizing the data and minimizing redundant data. Rather than storing data in separate files for each application, data are stored physically to appear to users as being stored in only one location. A single database services multiple applications. For example, instead of a corporation storing employee data in separate information systems and separate files for personnel, payroll, and benefits, the corporation could create a single common human resources database.

4.5.1 Database Management Systems
A database management system (DBMS) is simply the software that permits an organization to centralize data, manage them efficiently, and provide access to the stored data by application programs. The DBMS acts as an interface between application programs and the physical data files. When the application program calls for a data item such as gross pay, the DBMS finds this item in the database and presents it to the application program. Using traditional data files the programmer would have to specify the size and format of each data element used in the program and then tell the computer where they were located. A DBMS eliminates most of the data definition statements found in traditional programs.

The DBMS relieves the programmer or end user from the task of understanding where and how the data are actually stored by separating the logical and physical views of the data. The logical view presents data as they would be perceived by end users or business specialists, whereas the physical view shows how data are actually organized and structured on physical storage media. There is only one physical view of the data, but there can be many different logical views. The database management software makes the physical database available for different logical views presented for various application programs.

A database management system has three components:

1) A data definition language

2) A data manipulation language

3) A data dictionary

The data definition language is the formal language programmers use to specify the content and structure of the database. The data definition language defines each data element as it appears in the database before that data element is translated into the forms required by application programs.

Most DBMS have a specialized language called a data manipulation language that is used in conjunction with some conventional third- or fourth-generation programming languages to manipulate the data in the database. This language contains commands that permit end users and programming specialists to extract data from the database to satisfy information requests and develop applications. The most prominent data manipulation language today is Structured Query Language, or SQL. End users and information systems specialists can use SQL as an interactive query language to access data from databases, and SQL commands can be embedded in application programs written in conventional programming languages.

The third element of a DBMS is a data dictionary. This is an automated or manual file that stores definitions of data elements and data characteristics such as usage, physical representation, ownership (who in the organization is responsible for maintaining the data), authorization, and security. Many data dictionaries can produce lists and reports of data use, groupings, program locations, and so on. A data element represents a field.

By creating an inventory of data contained in the database, the data dictionary serves as an important data management tool. For instance, business users could consult the dictionary to find out exactly what pieces of data are maintained for the sales or marketing function or even to determine all the information maintained by the entire enterprise. The dictionary could supply business users with the name, format, and specifications required to access data for reports. Technical staff could use the dictionary to determine what data elements and files must be changed if a program is changed.

Most data dictionaries are entirely passive; they simply report. More advanced types are active; changes in the dictionary can be automatically used by related programs. For instance, to change ZIP codes from five to nine digits, one could simply enter the change in the dictionary without having to modify and recompile all application programs using ZIP codes.

In an ideal database environment, the data in the database are defined only once and used for all applications whose data reside in the database, thereby eliminating data redundancy and inconsistency. Application programs, which are written using a combination of the data manipulation language of the DBMS and a conventional programming language, request data elements from the database. Data elements called for by the application programs are found and delivered by the DBMS. The programmer does not have to specify in detail how or where the data are to be found.

A DBMS can reduce program-data dependence along with program development and maintenance costs. Access and availability of information can be increased because users and programmers can perform ad hoc queries of data in the database. The DBMS allows the organization to centrally manage data, their use, and security.

4.6 Types of Databases

Contemporary DBMS use different database models to keep track of entities, attributes, and relationships. Each model has certain processing advantages and certain business advantages.

4.6.1Relational DBMS

The most popular type of DBMS today for PCs as well as for larger computers and mainframes is the relational DBMS. The relational data model represents all data in the database as simple two-dimensional tables called relations. The tables appear similar to flat files, but the information in more than one file can be easily extracted and combined. Sometimes the tables are referred to as files.

In a relational database, three basic operations as shown in Figure 4-3 are used to develop useful sets of data: **select, project, and join.**

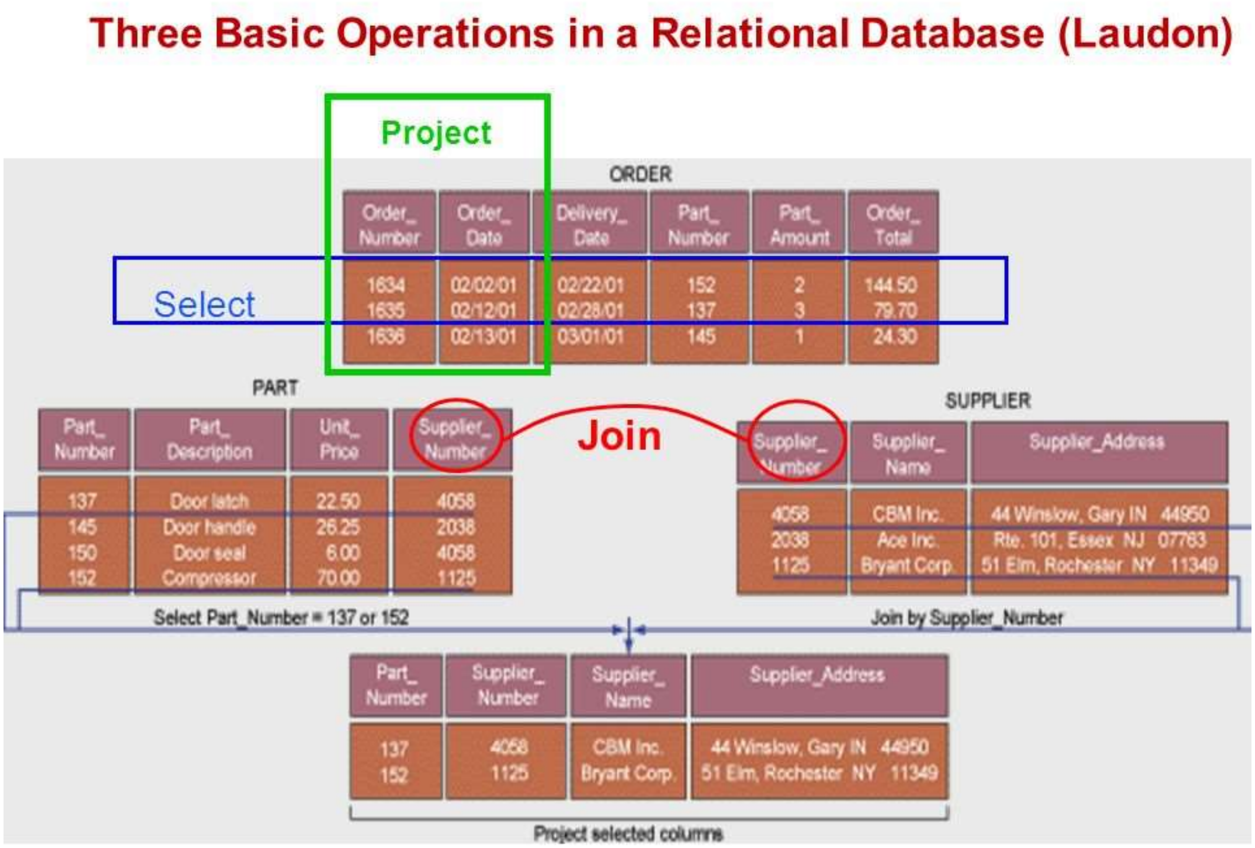

Figure 4-3

The **select** operation creates a subset consisting of all records in the file that meet stated criteria. Select creates, in other words, a subset of rows that meet certain criteria. In our example, we want to select records (rows) from the part table where the part number equals 137 or 152.

The join operation combines relational tables to provide the user with more information than is available in individual tables. In our example we want to join the now-shortened part table (only parts numbered 137 or 152 will be presented) and the supplier The project operation creates a subset consisting of columns in a table, permitting the user to create new tables (also called views) that contain only the information required. In our example, we want to extract from the new result table only the following columns: Part_Number, Supplier_Number, Supplier_Name, and Supplier_Address.

The SQL statements for producing the new result table in Figure 4-3 would be:

SELECT PART.Part_Number, SUPPLIER.Supplier_Number,

SUPPLIER.Supplier_Name,

SUPPLIER.Supplier_Address

FROM PART, SUPPLIER

WHERE PART.Supplier_Number = SUPPLIER.Supplier_Number AND

Part_Number = 137

OR

Part_Number = 152;

Leading mainframe relational database management systems include IBM's DB2 and Oracle from the Oracle Corporation. DB2, Oracle, and Microsoft SQL Server are used as DBMS for midrange computers. Microsoft Access is a PC relational database management system, and Oracle Lite is a DBMS for small, handheld computing devices.

4.6.2Hierarchical and Network DBMS

One can still find older systems that are based on a hierarchical or network data model. The hierarchical DBMS presents data to users in a treelike structure. Within each record, data elements are organized into pieces of records called segments. To the user, each record looks like an organization chart with one top-level segment called the root. An upper segment is connected logically to a lower segment in a parentchild relationship. A parent segment can have more than one child, but a child can have only one parent.

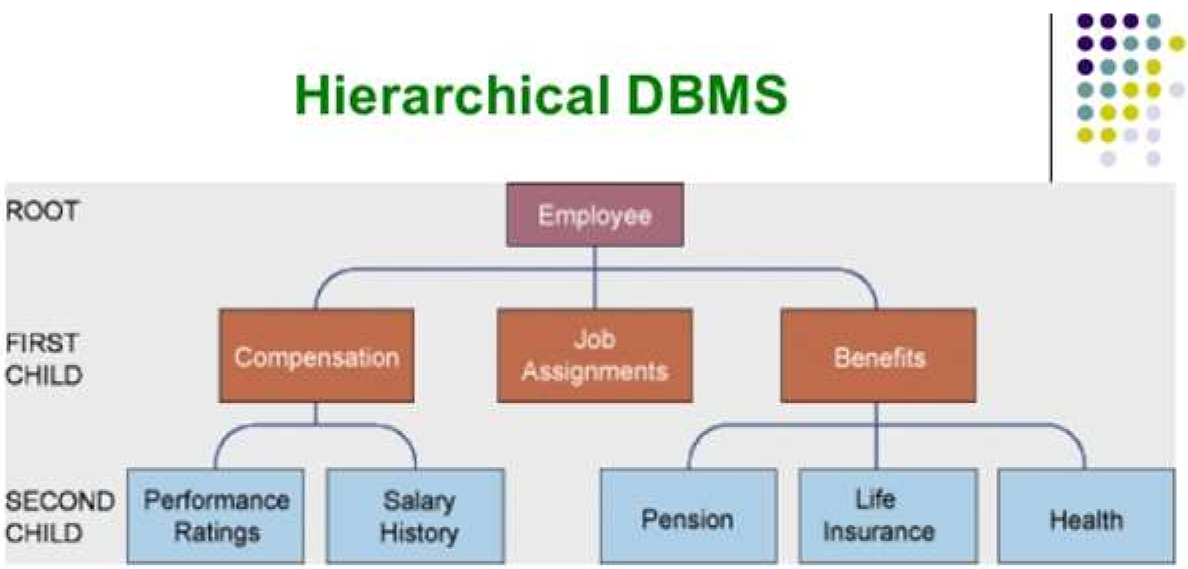

Figure 4-4

Figure 4-4 shows a hierarchical structure . The root segment is Employee, which contains basic employee information such as name, address, and identification number. Immediately below it are three child segments: Compensation (containing salary and promotion data), Job Assignments (containing data about job positions and departments), and Benefits (containing data about beneficiaries and benefit options). The Compensation segment has two children below it: Performance Ratings (containing data about employees' job performance evaluations) and Salary History (containing historical data about employees' past salaries). Below the Benefits segment are child segments for Pension, Life Insurance, and Health, containing data about these benefit plans.

Whereas hierarchical structures depict one-to-many relationships, network DBMS depict data logically as many-to-many relationships. In other words, parents can have multiple children, and a child can have more than one parent. A typical many-to-many relationship for a network DBMS is the student course relationship . There are many courses in a university and many students. A student takes many courses and a course has many students.

Hierarchical and network DBMS are considered outdated and are no longer used for building new database applications. They are much less flexible than relational DBMS and do not support ad hoc, English language-like inquiries for information. All paths for accessing data must be specified in advance and cannot be changed without a major programming effort.

Relational DBMS, in contrast have much more flexibility in providing data for ad hoc queries, combining information from different sources, and providing capability to add new data and records without disturbing existing programs and applications. However, these systems can be slowed down if they require many accesses to the data stored on disk to carry out the select, join, and project commands. Selecting one part number from among millions, one record at a time, can take a long time. Of course the database can be tuned to speed up prespecified queries.

Hierarchical DBMS can still be found in large legacy systems that require intensive high-volume transaction processing. A legacy system is a system that has been in existence for a long time and that continues to be used to avoid the high cost of replacing or redesigning it. Banks, insurance companies, and other high-volume users continue to use reliable hierarchical databases such as IBM's IMS (Information Management System), developed in 1969. Many organizations have converted to DB2, IBM's relational DBMS for new applications, while retaining IMS for traditional transaction processing. For example, Dallas-based Texas Instruments depends on IMS for its heavy processing requirements, including inventory, accounting, and manufacturing. As relational products acquire more muscle, firms will shift away completely from hierarchical DBMS, but this will happen over a long period of time.

4.6.3 Object-Oriented Databases

Conventional database management systems were designed for homogeneous data that can be easily structured into predefined data fields and records organized in rows or tables. But many applications today and in the future will require databases that can store and retrieve not only structured numbers and characters but also drawings, images, photographs, voice, and full-motion video. Conventional DBMS are not well suited to handling graphics-based or multimedia applications. For instance, design data in a computer-aided design (CAD) database consist of complex relationships among many types of data. Manipulating these kinds of data in a relational system requires extensive programming to translate these complex data structures into tables and rows. An object-oriented DBMS, however, stores the data and procedures as objects that can be automatically retrieved and shared.

Object-oriented database management systems (OODBMS) are becoming popular because they can be used to manage the various multimedia components or Java applets used in Web applications, which typically integrate pieces of information from a variety of sources. OODBMS also are useful for storing data types such as recursive data. (An example would be parts within parts as found in manufacturing applications.) Finance and trading applications often use OODBMS because they require data models that must be easy to change to respond to new economic conditions.

Although object-oriented databases can store more complex types of information than relational DBMS, they are relatively slow compared with relational DBMS for processing large numbers of transactions. Hybrid object-relational DBMS systems are now available to provide capabilities of both object-oriented and relational DBMS. A hybrid approach can be accomplished in three different ways: by using tools that offer object-oriented access to relational DBMS, by using object-oriented extensions to existing relational DBMS, or by using a hybrid object-relational database management system.

4.7 Creating a Database Environment

In order to create a database environment, one must understand the relationships among the data, the type of data that will be maintained in the database, how the data will be used, and how the organization will need to change to manage data from a company-wide perspective. We now describe important database design principles and the management and organizational requirements of a database environment.

4.7.1Designing Databases

To create a database, one must go through two design exercises: a conceptual design and a physical design. The conceptual, or logical, design of a database is an abstract model of the database from a business perspective, whereas the physical design shows how the database is actually arranged on direct access storage devices. Logical design requires a detailed description of the business information needs of the actual end users of the database. Ideally, database design will be part of an overall organizational data planning effort .

The conceptual database design describes how the data elements in the database are to be grouped. The design process identifies relationships among data elements and the most efficient way of grouping data elements together to meet information requirements. The process also identifies redundant data elements and the groupings of data elements required for specific application programs. Groups of data are organized, refined, and streamlined until an overall logical view of the relationships among all the data elements in the database emerges.

To use a relational database model effectively, complex groupings of data must be streamlined to eliminate redundant data elements and awkward many-to-many relationships. The process of creating small, stable data structures from complex groups of data is called normalization. In the particular business modeled here, an order can have more than one part but each part is provided by only one supplier. If we built a relation called ORDER with all the fields included here, we would have to repeat the name, description, and price of each part on the order and the name and address of each part vendor. This relation contains what are called repeating groups because there can be many parts and suppliers for each order, and it actually describes multiple entities—parts and suppliers as well as orders. A more efficient way to arrange the data is to break down ORDER into smaller relations, each of which describes a single entity.

If a database has been carefully considered, with a clear understanding of business information needs and usage, the database model will most likely be in some normalized form. Many real-world databases are not fully normalized because this may not be the most sensible way to meet business information requirements

4.7.2Distributing Databases

Database design also considers how the data are to be distributed. Information systems can be designed with a centralized database that is used by a single central processor or by multiple processors in a client/server network. Alternatively, the database can be distributed. A distributed database is one that is stored in more than one physical location. Parts of the database are stored physically in one location, and other parts are stored and maintained in other locations. There are two main ways of distributing a database . The central database can be partitioned so that each remote processor has the necessary data to serve its local area. Changes in local files can be justified with the central database on a batch basis, often at night. Another strategy is to replicate the central database at all remote locations.

Distributed systems reduce the vulnerability of a single, massive central site. They increase service and responsiveness to local users and often can run on smaller less expensive computers. Distributed systems, however, are dependent on high-quality telecommunications lines, which themselves are vulnerable. Moreover, local databases can sometimes depart from central data standards and definitions, and they pose security problems by widely distributing access to sensitive data. Database designers need to weigh these factors in their decisions.

4.8Management Requirements for Database Systems

Much more is required for the development of database systems than simply selecting a logical database model. The database is an organizational discipline, a method, rather than a tool or technology. It requires organizational and conceptual change. Without management support and understanding, database efforts fail. The critical elements in a database environment are

(1) data administration,

(2) data planning and modeling methodology,

(3) database technology and management, and

(4) users.

4.8.1 Data Administration

Database systems require that the organization recognize the strategic role of information and begin actively to manage and plan for information as a corporate resource. This means that the organization must develop a data administration function with the power to define information requirements for the entire company and with direct access to senior management. The chief information officer (CIO) or vice president of information becomes the primary advocate in the organization for database systems.

Data administration is responsible for the specific policies and procedures through which data can be managed as an organizational resource. These responsibilities include developing information policy, planning for data, overseeing logical database design and data dictionary development, and monitoring how information system specialists and end-user groups use data.

The fundamental principle of data administration is that all data are the property of the organization as a whole. Data cannot belong exclusively to any one business area or organizational unit. All data are to be made available to any group that requires them to fulfill its mission. An organization needs to formulate an information policy that specifies its rules for sharing, disseminating, acquiring, standardizing, classifying, and inventorying information throughout the organization. Information policy lays out specific procedures and accountabilities, specifying which organizational units share information, where information can be distributed, and who has responsibility for updating and maintaining the information. Although data administration is a very important organizational function, it has proved very challenging to implement.

4.8.2Data Planning and Modeling Methodology

The organizational interests served by the DBMS are much broader than those in the traditional file environment; therefore, the organization requires enterprise-wide planning for data. Enterprise analysis, which addresses the information requirements of the entire organization (as opposed to the requirements of individual applications), is needed to develop databases. The purpose of enterprise analysis is to identify the key entities, attributes, and relationships that constitute the organization's data.

4.8.3Database Technology, Management, and Users

Databases require new software and a new staff specially trained in DBMS techniques, as well as new data management structures. Most corporations develop a database design and management group within the corporate information system division that is responsible for defining and organizing the structure and content of the database and maintaining the database. In close cooperation with users, the design group establishes the physical database, the logical relations among elements, and the access rules and procedures. The functions it performs are called database administration.

A database serves a wider community of users than traditional systems. Relational systems with fourth-generation query languages permit employees who are not computer specialists to access large databases. In addition, users include trained computer specialists. To optimize access for no specialists, resources must be devoted to training end users.

4.9 Database Trends

Organizations are installing powerful data analysis tools and data warehouses to make better use of the information stored in their databases and are taking advantage of database technology linked to the World Wide Web. We now explore these developments.

4.9 Multidimensional Data Analysis

Sometimes managers need to analyze data in ways that traditional database models cannot represent. For example, a company selling four different products—nuts, bolts, washers, and screws—in the East, West, and Central regions, might want to know actual sales by product for each region and might also want to compare them with projected sales. This analysis requires a multidimensional view of data.

To provide this type of information, organizations can use either a specialized multidimensional database or a tool that creates multidimensional views of data in relational databases. Multidimensional analysis enables users to view the same data in different ways using multiple dimensions. Each aspect of information—product, pricing, cost, region, or time period—represents a different dimension. So a product manager could use a multidimensional data analysis tool to learn how many washers were sold in the East in June, how that compares with the previous month and the previous June, and how it compares with the sales forecast. Another term for multidimensional data analysis is on-line analytical processing (OLAP).

4.10Data Warehouses and Datamining

Decision makers need concise, reliable information about current operations, trends, and changes. What has been immediately available at most firms is current data only (historical data were available through special IS reports that took a long time to produce). Data often are fragmented in separate operational systems, such as sales or payroll, so that different managers make decisions from incomplete knowledge bases. Users and information system specialists may have to spend inordinate amounts of time locating and gathering data (Watson and Haley, 1998). Data warehousing addresses this problem by integrating key operational data from around the company in a form that is consistent, reliable, and easily available for reporting.

4.10.1 What Is a Data Warehouse?

A data warehouse is a database that stores current and historical data of potential interest to managers throughout the company. The data originate in many core operational systems and external sources, including Web site transactions, each with different data models. They may include legacy systems, relational or object-oriented DBMS applications, and systems based on HTML or XML documents. The data from these diverse applications are copied into the data warehouse database as often as needed—hourly, daily, weekly, monthly. The data are standardized into a common data model and consolidated so that they can be used across the enterprise for management analysis and decision making. The data are available for anyone to access as needed but cannot be altered.

Companies can build enterprise-wide data warehouses where a central data warehouse serves the entire organization, or they can create smaller, decentralized warehouses called data marts. A data mart is a subset of a data warehouse in which a summarized or highly focused portion of the organization's data is placed in a separate database for a specific population of users. For example, a company might develop marketing and sales data marts to deal with customer information. A data mart typically focuses on a single subject area or line of business, so it usually can be constructed more rapidly and at lower cost than an enterprise-wide data warehouse. However, complexity, costs, and management problems will rise if an organization creates too many data marts.

4.10.2Data mining

A data warehouse system provides a range of ad hoc and standardized query tools, analytical tools, and graphical reporting facilities, including tools for OLAP and data mining. Data mining uses a variety of techniques to find hidden patterns and relationships in large pools of data and infer rules from them that can be used to predict future behavior and guide decision making . Data mining helps companies engage in one-to-one marketing where personalized or individualized messages can be created based on individual preferences Data mining is both a powerful and profitable tool, but it poses challenges to the protection of individual privacy. Data mining technology can combine information from many diverse sources to create a detailed "data image" about each of us—our income, our driving habits, our hobbies, our families, and our political interests.

4.10.3Benefits of Data Warehouses

Data warehouses not only offer improved information but they also make it easy for decision makers to obtain it. They even include the ability to model and remodel the data. It has been estimated that 70 percent of the world's business information resides on mainframe databases, many of which are for older legacy systems. Many of these legacy systems are critical production applications that support the company's core business processes. As long as these systems can efficiently process the necessary volume of transactions to keep the company running, firms are reluctant to replace them to avoid disrupting critical business functions and high system replacement costs. Many of these legacy systems use hierarchical DBMS where information is difficult for users to access. Data warehouses enable decision makers to access data as often as they need without affecting the performance of the underlying operational systems. Many organizations are making access to their data warehouses even easier by using Web technology.

4.11Databases and the Web

Database technology plays an important role in making organizations' information resources available on the World Wide Web. We now explore the role of hypermedia databases in the Web and the growing use of Web sites to access information stored in conventional databases inside the firm.

4.11.1The Web and Hypermedia Databases

Web sites store information as interconnected pages containing text, sound, video, and graphics using a hypermedia database. The hypermedia database approach to information management stores chunks of information in the form of nodes connected by links the user specifies . The nodes can contain text, graphics, sound, full-motion video, or executable computer programs. Searching for information does not have to follow a predetermined organization scheme. Instead, one can branch instantly to related information in any kind of relationship the author establishes. The relationship between records is less structured than in a traditional DBMS.

The hypermedia database approach enables users to access topics on a Web site in whatever order they wish. For instance, from the Web page from the U.S. National Oceanic and Atmospheric Administration (NOAA) illustrated on this page, one could branch to other Web pages by clicking on the topics in the left column. In addition to welcoming visitors to NOAA, these Web pages provide more information on NURP (National Undersea Research Program), News from the Deep, Undersea Research Centers, Funding Opportunities, Education, Research Highlights and Products, Undersea Technologies, and Undersea Web sites. The links from the on-screen page to the other related Web pages are highlighted in blue.

4.11.2Linking Internal Databases to the Web

A series of middleware and other software products has been developed to help users gain access to organizations' legacy data through the Web. For example, a customer with a Web browser might want to search an on-line retailer's database for pricing information. The user would access the retailer's Web site over the Internet using Web browser software on his or her client PC. The user's Web browser software would request data from the organization's database, using HTML commands to communicate with the Web server. Because many back-end databases cannot interpret commands written in HTML, the Web server would pass these requests for data to special software that would translate HTML commands into SQL so that they could be processed by the DBMS working with the database. In a client/server environment, the DBMS resides on a special dedicated computer called a database server. The DBMS receives the SQL requests and provides the required data. The middleware would transfer information from the organization's internal database back to the Web server for delivery in the form of a Web page to the user.

4.12SUMMARY

Selecting an appropriate data model and data management technology for the organization is a key management decision. Managers will need to evaluate the costs and benefits of implementing a database environment and the capabilities of various DBMS or file management technologies. Management should ascertain that organizational databases are designed to meet management information objectives and the organization's business needs.

The organization's data model should reflect its key business processes and decision-making requirements. Data planning may need to be performed to make sure that the organization's data model delivers information efficiently for its business processes and enhances organizational performance. Designing a database is an organizational endeavor.

Multiple database and file management options are available for organizing and storing information. Key technology decisions should consider the efficiency of accessing information, flexibility in organizing information, the type of information to be stored and arranged, compatibility with the organization's data model, and compatibility with the organization's hardware and operating systems.

4.13 Questions for Exercise

Q1) Discuss the term data organization. How data are organized in the traditional file environment. Discuss the problems with the traditional file environment.

Q2) Define DBMS and discuss their components.

Q3) Differentiate between Relational DBMS, Hierarchical and Network DBMS.

Q4) Define Object-Oriented Databases.

Q5) Discuss the elements in a database environment.

Q6) Define the term Data Warehouse and Data Mining . Discuss the benefits of Data Warehouse.

Q7) How the Web and the Hypermedia Databases are related to each other.

4.14 Further Readings

1. Information Technology for management, Turban, McLean, Wetherbe, 4th edition, Wiley

2. Management Information Systems, Loudon and Loudon, 10th edition, Pearsons Educations

3. Management Information Systems, Jaswal Oxford Press

4.J. O'Brian, Management Information Systems: Managing Information Technology in the Networked Enterprise (3rd Ed), Irwin, 1996.

5. Robert Schultheis & Mary Sumner, Management Information Systems-The Manager's View, Tata McGraw Hill, New Delhi