Chapter 6

Database Management

6.1 Hierarchy of Data [Figure 6.1][Slide 6-4]

Data are the principal resources of an organization. Data stored in computer systems form a hierarchy extending from a single bit to a database, the major record-keeping entity of a firm. Each higher rung of this hierarchy is organized from the components below it.

Data are logically organized into:

1. Bits (characters)

2. Fields

3. Records

4. Files

5. Databases

Bit (Character) - a bit is the smallest unit of data representation (value of a bit may be a 0 or 1). Eight bits make a byte which can represent a character or a special symbol in a character code.

Field - a field consists of a grouping of characters. A data field represents an attribute (a characteristic or quality) of some entity (object, person, place, or event).

Record - a record represents a collection of attributes that describe a real-world entity. A record consists of fields, with each field describing an attribute of the entity.

File - a group of related records. Files are frequently classified by the application for which they are primarily used (employee file). A ***primary key*** in a file is the field (or fields) whose value identifies a record among others in a data file.

Database - is an integrated collection of logically related records or files. A database consolidates records previously stored in separate files into a common pool of data records that provides data for many applications. The data is managed by systems software called database management systems (DBMS). The data stored in a database is independent of the application programs using it and of the types of secondary storage devices on which it is stored.

6.2 File Environment and its Limitations

There are three principal methods of organizing files, of which only two provide the direct access necessary in on-line systems.

File Organization [Figure 6.2 & 6.3]

Data files are organized so as to facilitate access to records and to ensure their efficient storage. A tradeoff between these two requirements generally exists: if rapid access is required, more storage is required to make it possible.

Access to a record for reading it is the essential operation on data. There are two types of access:

1. ***Sequential access*** - is performed when records are accessed in the order they are stored. Sequential access is the main access mode only in batch systems, where files are used and updated at regular intervals.

2. ***Direct access*** - on-line processing requires direct access, whereby a record can be accessed without accessing the records between it and the beginning of the file. The primary key serves to identify the needed record.

There are three methods of file organization: [Table 6.1]

1. Sequential organization

2. Indexed-sequential organization

3. Direct organization

Sequential Organization

In sequential organization records are physically stored in a specified order according to a key field in each record.

Advantages of sequential access:

1. It is fast and efficient when dealing with large volumes of data that need to be processed periodically (batch system).

Disadvantages of sequential access:

1. Requires that all new transactions be sorted into the proper sequence for sequential access processing.

2. Locating, storing, modifying, deleting, or adding records in the file requires rearranging the file.

3. This method is too slow to handle applications requiring immediate updating or responses.

Indexed-Sequential Organization

In the indexed-sequential files method, records are physically stored in sequential order on a magnetic disk or other direct access storage device based on the key field of each record. Each file contains an index that references one or more key fields of each data record to its storage location address.

Direct Organization

Direct file organization provides the fastest direct access to records. When using direct access methods, records do not have to be arranged in any particular sequence on storage media. Characteristics of the direct access method include:

1. Computers must keep track of the storage location of each record using a variety of direct organization methods so that data can be retrieved when needed.

2. New transactions' data do not have to be sorted.

3. Processing that requires immediate responses or updating is easily performed.

6.3 Database Environment [Figure 6.6][Slide 6-5]

A database is an organized collection of interrelated data that serves a number of applications in an enterprise. The database stores not only the values of the attributes of various entities but also the relationships between these entities. A database is managed by a database management system (DBMS), a systems software that provides assistance in managing databases shared by many users.

A DBMS:

1. Helps organize data for effective access by a variety of users with different access needs and for efficient storage.

2. It makes it possible to create, access, maintain, and control databases.

3. Through a DBMS, data can be integrated and presented on demand.

Advantages of a database management approach:

1. Avoiding uncontrolled data redundancy and preventing inconsistency

2. Program-data independence

3. Flexible access to shared data

4. Advantages of centralized control of data

6.4 Levels of Data Definition in Databases [Figure 6.7]

The user view of a DBMS becomes the basis for the date modelling steps where the relationships between data elements are identified. These data models define the logical relationships among the data elements needed to support a basic business process. A DBMS serves as a logical framework (schema, subschema, and physical) on which to base the physical design of databases and the development of application programs to support the business processes of the organization. A DBMS enables us to define a database on three levels:

1. ***Schema*** - is an overall logical view of the relationships between data in a database.

2.***Subschema*** - is a logical view of data relationships needed to support specific end user application programs that will access the database.

3.***Physical*** - looks at how data is physically arranged, stored, and accessed on the magnetic disks and other secondary storage devices of a computer system.

A DBMS provides the language, called ***data definition language*** (DDL), for defining the database objects on the three levels. It also provides a language for manipulating the data, called the ***data manipulation language*** (DML), which makes it possible to access records, change values of attributes, and delete or insert records.

6.5 Data Models or How to Represent Relationships between Data

A data model is a method for organizing databases on the logical level, the level of the schema and subschemas. The main concern in such a model is how to represent relationships among database records. The relationships among the many individual records in databases are based on one of several logical data structures or models. DBMS are designed to provide end users with quick, easy access to information stored in databases. Three principal models include:

1. Hierarchical Structure

2. Network Structure

3. Relational Structure

Hierarchical:

Early mainframe DBMS packages used the ***hierarchical structure***, in which:

1. Relationships between records form a hierarchy or tree like structure.

2. Records are dependent and arranged in multilevel structures, consisting of one root record & any number of subordinate levels.

3. Relationships among the records are one-to-many, since each data element is related only to one element above it.

4. Data element or record at the highest level of the hierarchy is called the root element. Any data element can be accessed by moving progressively downward from the root and along the branches of the tree until the desired record is located.

Network Structure:

The network structure:

1. Can represent more complex logical relationships, and is still used by many mainframe DBMS packages.

2. Allows many-to-many relationship among records. That is, the network model can access a data element by following one of several paths, because any data element or record can be related to any number of other data elements.

Relational Structure:

The relational structure:

1. Most popular of the three database structures.

2. Used by most microcomputer DBMS packages, as well as many minicomputer and mainframe systems.

3. Data elements within the database are stored in the form of simple tables. Tables are related if they contain common fields.

4. DBMS packages based on the relational model can link data elements from various tables to provide information to users.

Evaluation of Database Structures

MODEL	ADVANTAGES	DISADVANTAGES
Hierarchical Data Structure	Ease with which data can be stored and retrieved in structured,	Hierarchical one-to many relationships must be specified in

	routine types of transactions. Ease with which data can be extracted for reporting purposes. Routine types of transaction processing is fast and efficiently.	advance, and are not flexible. Cannot easily handle ad hoc requests for information. Modifying a hierarchical database structure is complex. Great deal of redundancy. Requires knowledge of a programming language.
Network Structure	More flexible that the hierarchical model. Ability to provide sophisticated logical relationships among the records	Network many-to-many relationships must be specified in advance User is limited to retrieving data that can be accessed using the established links between records. Cannot easily handle ad hoc requests for information. Requires knowledge of a programming language.
Relational Structure	Flexible in that it can handle ad hoc information requests. Easy for programmers to work with. End users can use this model with litter effort or training. Easier to maintain than the hierarchical and network models.	Cannot process large amounts of business transactions as quickly and efficiently as the hierarchical and network models.

6.6 Relational Databases [Figure 6.11, 6.13]

A relational database is a collection of tables. Such a database is relatively easy for end users to understand. Relational databases afford flexibility across the data and are easy to understand and modify.

1. Select, which selects from a specified table the rows that satisfy a given condition.

2. Project, which selects from a given table the specified attribute values

3. Join, which builds a new table from two specified tables.

The power of the relational model derives from the join operation. It is precisely because records are related to one another through a join operation, rather than through links, that we do not need a predefined access path. The join operation is also highly time-consuming, requiring access to many records stored on disk in order to find the needed records.

6.7 SQL - A Relational Query Language

Structured Query Languages (SQL) has become an international standard access language for defining and manipulating data in databases. It is a data-definition-and-management language of most well-known DBMS, including some nonrelational ones. SQL may be used as an independent query language to define the objects in a database, enter the data into the database, and access the data. The so-called embedded SQL is also provided for programming in procedural languages (Ahost@ languages), such as C, COBOL, or PL/L, in order to access a database from an application program. In the end-user environment, SQL is generally hidden by more user-friendly interfaces.

The principal facilities of SQL include:

1. Data definition

2. Data manipulation

6.8 Designing a Relational Database

Database design progresses from the design of the logical levels of the schema and the subschema to the design of the physical level.

The aim of ***logical design***, also known as ***data modeling***, is to design the schema of the database and all the necessary subschemas. A relational database will consist of tables (relations), each of which describes only the

attributes of a particular class of entities. Logical design begins with identifying the entity classes to be represented in the database and establishing relationships between pairs of these entities. A relationship is simply an interaction between the entities represented by the data. This relationship will be important for accessing the data. Frequently, ***entity-relationship (E-R) diagrams***, are used to perform data modeling.

Normalization is the simplification of the logical view of data in relational databases. Each table is normalized, which means that all its fields will contain single data elements, all its records will be distinct, and each table will describe only a single class of entities. The objective of normalization is to prevent replication of data, with all its negative consequences.

After the logical design comes the ***physical design*** of the database. All fields are specified as to their length and the nature of the data (numeric, characters, and so on). A principal objective of physical design is to minimize the number of time-consuming disk accesses that will be necessary in order to answer typical database queries. Frequently, indexes are provided to ensure fast access for such queries.

6.9 The Data Dictionary

A ***data dictionary*** is a software module and database containing descriptions and definitions concerning the structure, data elements, interrelationships, and other characteristics of an organization's database.

Data dictionaries store the following information about the data maintained in databases:

1. Schema, subschemas, and physical schema

2. Which applications and users may retrieve the specific data and which applications and users are able to modify the data

3. Cross-reference information, such as which programs use what data and which users receive what reports

4. Where individual data elements originate, and who is responsible for maintaining the data

5. What the standard naming conventions is for database entities.

6. What the integrity rules is for the data

7. Where the data are stored in geographically distributed databases.

A data dictionary:

1. Contains all the data definitions, and the information necessary to identify data ownership

2. Ensures security and privacy of the data, as well as the information used during the development and maintenance of applications which rely on the database.

6.10 Managing the Data Resource of an Organization

The use of database technology enables organizations to control their data as a resource, however, it does not automatically produce organizational control of data.

Components of Information Resource Management [Figure 6.17]

Both organizational actions and technological means are necessary to:

1. Ensure that a firm systematically accumulates data in its databases

2. Maintains the data over time

3. Provides the appropriate access to the data to the appropriate employees.

The principal components of this information resource management are:

1. Organizational processes

- Information Planning and data modeling

2. Enabling technologies

- DBMS and a Data Dictionary

3. Organizational functions

- data administration and database administration

Database Administration and Database Administration [Figure 6.18]

The functional units responsible for managing the data are:

1. Data administrator (DA)

2. Database administrator (DBA)

Data administrator - the person who has the central responsibility for an organizations data.

Responsibilities include:

1. Establishing the policies and specific procedures for collecting, validating, sharing, and inventorying data to be stored in databases and for making information accessible to the members of the organization and, possibly, to persons outside of it.

2. Data administration is a policy making function and the DA should have access to senior corporate management.

3. Key person involved in the strategic planning of the data resource.

4. Often defines the principal data entities, their attributes, and the relationships among them.

Database Administrator - is a specialist responsible for maintaining standards for the development, maintenance, and security of an organization's databases.

Responsibilities include:

1. Creating the databases and carrying out the policies laid down by the data administrator.

2. In large organizations, the DBA function is actually performed by a group of professionals. In a small firm, a programmer/analyst may perform the DBA function, while one of the managers acts as the DA.

3. Schema and subschemas of the database are most often defined by the DBA, who has the requisite technical knowledge. They also define the physical layout of the databases, with a view toward optimizing system performance for the expected pattern of database usage.

Joint responsibilities of the DA and DBA:

1. Maintaining the data dictionary
2. Standardizing names and other aspects of data definition
3. Providing backup
4. Provide security for the data stored in a database, and ensure privacy based on this security.
5. Establish a disaster recovery plan for the databases

6.11 Developmental Trends in Database Management

Three important trends in database management include:

1. Distributed databases
2. Data warehousing
3. Rich databases (includes object-oriented databases)

Distributed Databases [Figure 6.19][Slide 6-8]

Distributed databases are that are spread across several physical locations. In distributed databases, the data are placed where they are used most often, but the entire database is available to each authorized user. These are databases of local work groups (LAN), and departments at regional offices (WAN), branch offices, manufacturing plants, and other work sites. These databases can include segments of both common operational and common user databases, as well as data generated and used only at a user's own site.

Data Warehouses Databases [Figure 6.20]

A data warehouse stores data from current and previous years that has been extracted from the various operational and management databases of an organization. It is a central source of data that has been standardized and integrated so it can be used by managers and other end user professionals from throughout an organization. The objective of a corporate data warehouse is to continually select data from the operational databases, transform the data into a uniform format, and open the warehouse to the end users through a friendly and consistent interface.

Data warehouses are also used for data mining - automated discovery of potentially significant relationships among various categories of data.

Systems supporting a data warehouse consists of three components:

1. Extract and Prepare Data

- the first subsystem extracts the data from the operational systems, many of them older legacy systems, and Ascrubs@ it by removing errors and inconsistencies.

2. Store Date in the Warehouse

- the second support component is actually the DBMS that will manage the warehouse data.

3. Provide Access and Analysis Capabilities

- the third subsystem is made up of the query tools that help users access the data and includes the OLAP and other DSS tools supporting data analysis.

Object-oriented and other Rich Databases

With the vastly expanded capabilities of information technology, the content of the databases is becoming richer. Traditional databases have been oriented toward largely numerical data or short fragments of text, organized into well-structured records. As the processing and storage capabilities of computer systems expand and as the telecommunications capacities grow, it is possible to support knowledge work more fully with rich data. These include:

1. Geographic information systems
2. Object-oriented databases
3. Hypertext and hypermedia databases
4. Image databases and text databases