TYPES OF DBMS

5.1 INTRODUCTION: There are four main types of database management systems (DBMS) and these are based upon their management of database structures. In other words, the **types of DBMS** are entirely dependent upon how the database is structured by that particular DBMS.

5.2 TYPES

Hierarchical DBMS

A DBMS is said to be hierarchical if the relationships among data in the database are established in such a way that one data item is present as the subordinate of another one or a sub unit. Here subordinate means that items have "parent-child" relationships among them. Direct relationships exist between any two records that are stored consecutively. The data structure "tree" is followed by the DBMS to structure the database. No backward movement is possible/allowed in the hierarchical database.

The hierarchical data model was developed by IBM in 1968 and introduced in information management systems. This model is like a structure of a tree with the records forming the nodes.

History

The hierarchical structure was used in early mainframe DBMS. Records' relationships form a treelike model. This structure is simple but inflexible because the relationship is confined to a one-to-many relationship. The IBM Information Management System (IMS) and the RDM Mobile are examples of a hierarchical database system with multiple hierarchies over the same data. RDM Mobile is a newly designed embedded database for a mobile computer system.

The hierarchical data model lost traction as Codd's relational model became the de facto standard used by virtually all mainstream database management systems. A relational-database implementation of a hierarchical model was first discussed in published form in 1992 (see also nested set model). Hierarchical data organization schemes resurfaced with the advent of XML in the late 1990s (see also XML database). The hierarchical structure is used primarily today for storing geographic information and file systems.

Currently hierarchical databases are still widely used especially in applications that require very high performance and availability such as banking and telecommunications. One of the most widely used commercial hierarchical databases is IMS. Another example of the use of hierarchical databases is Windows Registry in the Microsoft Windows operating systems.

Examples of hierarchical data represented as relational tables

An organization could store employee information in a table that contains attributes/columns such as employee number, first name, last name, and Department number. The organization provides each employee with computer hardware as needed, but computer equipment may only be used by the employee to which it is assigned. The organization could store the computer hardware information in a separate table that includes each part's serial number, type, and the employee that uses it. The tables might look like this:

In this model, the employee data table represents the "parent" part of the hierarchy, while the computer table represents the "child" part of the hierarchy. In contrast to tree structures usually found in computer software algorithms, in this model the children point to the parents. Each employee may possess several pieces of computer equipment, but each individual piece of computer equipment may have only one employee owner.

In this, the "child" is the same type as the "parent". The hierarchy stating EmpNo 10 is boss of 20, and 30 and 40 each report to 20 is represented by the "ReportsTo" column. In Relational database terms, the ReportsTo column is a foreign key referencing the EmpNo column. If the "child" data type were different, it would be in a different table, but there would still be a foreign key referencing the EmpNo column of the employees table.

This simple model is commonly known as the adjacency list model, and was introduced by Dr. Edgar F. Codd after initial criticisms surfaced that the relational model could not model hierarchical data.

The Windows Registry is a hierarchical database that stores configuration settings and options on Microsoft Windows operating systems.

Network DBMS

A DBMS is said to be a Network DBMS if the relationships among data in the database are of type many-to-many. The **many-to-many** communication paradigm

is one of three major Internet computing paradigms, characterized by multiple users contributing and receiving information, with the information elements often interlinked across different websites. Developments such as file sharing, blogs, Wikis, and tagging are media forms that reflect this paradigm; these contrast with both the one-to-one (characterized by e-mail, FTP, and Telnet) and one-to-many (characterized by websites) paradigms.

With the evolution to the full "many-to-many" computing paradigm, people can input and receive information to and from the Internet; they will be able to connect and communicate dynamically within a flexibly formed scope; there will be no artificial boundary between information and communication tools, and the definition of "many" will go well beyond people to include entities such as organizations, products, processes, events, and concepts.

The relationships among many-to-many appears in the form of a network. Thus the structure of a network database is extremely complicated because of these many-to-many relationships in which one record can be used as a key of the entire database. A network database is structured in the form of a graph that is also a data structure. Though the structure of such a DBMS is highly complicated however it has two basic elements i.e. records and sets to designate many-to-many relationships. Mainly high-level languages such as Pascal, C++, COBOL and FORTRAN etc. were used to implement the records and set structures.

Overview

While the hierarchical database model structures data as a tree of records, with each record having one parent record and many children, the network model allows each record to have multiple parent and child records, forming a generalized graph structure. This property applies at two levels: the schema is a generalized graph of record types connected by relationship types (called "set types" in CODASYL), and the database itself is a generalized graph of record occurrences connected by relationships (CODASYL "sets"). Cycles are permitted at both levels. The chief argument in favour of the network model, in comparison to the hierarchic model, was that it allowed a more natural modeling of relationships between entities. Although the model was widely implemented and used, it failed to become dominant for two main reasons. Firstly, IBM chose to stick to the hierarchical model with semi-network extensions in their established products such as IMS and DL/I. Secondly, it was eventually displaced by the relational model, which offered a higher-level, more declarative interface. Until the early 1980s the performance benefits of the low-level navigational interfaces offered by hierarchical and

network databases were persuasive for many large-scale applications, but as hardware became faster, the extra productivity and flexibility of the relational model led to the gradual obsolescence of the network model in corporate enterprise usage.

Database systems

Some well-known database systems that use the network model include:

- Integrated Data Store (IDS)
- IDMS (Integrated Database Management System)
- RDM Embedded
- RDM Server
- TurboIMAGE
- Univac DMS-1100

History

The network model's original inventor was Charles Bachman, and it was developed into a standard specification published in 1969 by the Conference on Data Systems Languages (CODASYL) Consortium. This was followed by a second publication in 1971, which became the basis for most implementations. Subsequent work continued into the early 1980s, culminating in an ISO specification, but this had little influence on products.

Relational DBMS

A DBMS is said to be a Relational DBMS or RDBMS if the database relationships are treated in the form of a table. There are three keys on relational DBMS: relation, domain and attributes. A network means it contains a fundamental constructs sets or records sets contains one to many relationship, records contains fields statical table that is composed of rows and columns is used to organize the database and its structure and is actually a two dimension array in the computer memory. A number of RDBMSs are available, some popular examples are Oracle, Sybase, Ingress, Informix, Microsoft SQL Server, and Microsoft Access.

Object-oriented DBMS

Able to handle many new data types, including graphics, photographs, audio, and video, object-oriented databases represent a significant advance over their other database cousins. Hierarchical and network databases are all designed to handle

structured data; that is, data that fits nicely into fields, rows, and columns. They are useful for handling small snippets of information such as names, addresses, zip codes, product numbers, and any kind of statistic or number you can think of. On the other hand, an object-oriented database can be used to store data from a variety of media sources, such as photographs and text, and produce work, as output, in a multimedia format.

- Object-oriented databases use small, reusable chunks of software called objects. The objects themselves are stored in the object-oriented database. Each object consists of two elements: 1) a piece of data (e.g., sound, video, text, or graphics), and 2) the instructions, or software programs called methods, for what to do with the data. Part two of this definition requires a little more explanation. The instructions contained within the object are used to do something with the data in the object. For example, test scores would be within the object as would the instructions for calculating average test score.

- Object-oriented databases have two disadvantages. First, they are more costly to develop. Second, most organizations are reluctant to abandon or convert from those databases that they have already invested money in developing and implementing. However, the benefits to object-oriented databases are compelling. The ability to mix and match reusable objects provides incredible multimedia capability. Healthcare organizations, for example, can store, track, and recall CAT scans, X-rays, electrocardiograms and many other forms of crucial data.

A **relational database management system** (**RDBMS**) is a database management system (DBMS) that is based on the relational model as invented by E. F. Codd, of IBM's San Jose Research Laboratory. Many popular databases currently in use are based on the relational database model.

RDBMSs have become a predominant choice for the storage of information in new databases used for financial records, manufacturing and logistical information, personnel data, and much more since the 1980s. Relational databases have often replaced legacy hierarchical databases and network databases because they are easier to understand and use. However, relational databases have been challenged by object databases, which were introduced

in an attempt to address the object-relational impedance mismatch in relational database, and XML databases

History

In 1974, IBM began developing System R, a research project to develop a prototype RDBMS.[4] Its first commercial product was SQL/DS, released in 1981.[5] However, the first commercially available RDBMS was Oracle, released in 1979 by Relational Software, now Oracle Corporation.[6] Other examples of an RDBMS include DB2, SAP Sybase ASE, and Informix.

Historical usage of the term

The term "relational database" was invented by E. F. Codd at IBM in 1970. Codd introduced the term in his seminal paper "A Relational Model of Data for Large Shared Data Banks". In this paper and later papers, he defined what he meant by "relational". One well-known definition of what constitutes a relational database system is composed of Codd's 12 rules. However, many of the early implementations of the relational model did not conform to all of Codd's rules, so the term gradually came to describe a broader class of database systems, which at a minimum:

- Present the data to the user as relations (a presentation in tabular form, i.e. as a *collection* of tables with each table consisting of a set of rows and columns);
- Provide relational operators to manipulate the data in tabular form.

The first systems that were relatively faithful implementations of the relational model were from the University of Michigan; Micro DBMS (1969), the Massachusetts Institute of Technology; (1971), and from IBM UK Scientific Centre at Peterlee; IS1 (1970–72) and its followon PRTV (1973–79). The first system sold as an RDBMS was Multics Relational Data Store, first sold in 1978. Others have been Berkeley Ingres QUEL and IBM BS12.

The most popular definition of an RDBMS is a product that presents a view of data as a collection of rows and columns, even if it is not based strictly upon relational theory. By this definition, RDBMS products typically implement some but not all of Codd's 12 rules.

A second school of thought argues that if a database does not implement all of Codd's rules (or the current understanding on the relational model, as expressed by Christopher J Date, Hugh Darwen and others), it is not relational. This view, shared by many theorists and other strict adherents to Codd's principles, would disqualify most DBMSs as not relational. For clarification, they often refer to some RDBMSs as *Truly-Relational Database Management Systems* (TRDBMS), naming others *Pseudo-Relational Database Management Systems* (PRDBMS).

As of 2009, most commercial relational DBMSes employ SQL as their query language. Alternative query languages have been proposed and implemented, notably the pre-1996 implementation of Berkeley Ingres QUEL.