****

****

毕业设计（论文）外文资料翻译

**学 院： 计算机科学与工程**

**专 业 班 级： 软件工程**

**指 导 教 师： 谷 钰**

**姓 名 学 号： 张智惠 20152550**

**外 文 出 处：** <http://www.bisheziliao.com/d-131036.html>

**2019年3月**

指导教师签字：

年 月 日

评阅人签字：

年 月 日

**作者：**Raghu Ramakrishnan

**英文原文：**

**Database Management Systems( 3th Edition ),Wiley ,2004, 5-12**

**An Introduction to Database Management System**

A database (sometimes spelled data base) is also called an electronic database, referring to any collection of data or information, and that is specially organized for rapid search and retrieval by a computer. Databases are structured to facilitate the storage, retrieval, modification, and deletion of data in conjunction with various data-processing operations .Databases can be stored on magnetic disk or tape, optical disk, or some other secondary storage device.

A database consists of a file or a set of files. The information in these files may be broken down into records, each of which consists of one or more fields. Fields are the basic units of data storage, and each field typically contains information pertaining to one aspect or attribute of the entity described by the database. Using keywords and various sorting commands, users can rapidly search, rearrange, group, and select the fields in many records to retrieve or create reports on particular aggregate of data.

Complex data relationships and linkages may be found in all but the simplest databases. The system software package that handles the difficult tasks associated with creating, accessing, and maintaining database records is called a database management system(DBMS). The programs in a DBMS package establish an interface between the database itself and the users of the database. (These users may be applications programmers, managers and others with information needs, and various OS programs)

A DBMS can organize, process, and present selected data elements form the database. This capability enables decision makers to search, probe, and query database contents in order to extract answers to nonrecurring and unplanned questions that aren’t available in regular reports. These questions might initially be vague and/or poorly defined, but people can “browse” through the database until they have the needed information. In short, the DBMS will “manage” the stored data items and assemble the needed items from the common database in response to the queries of those who aren’t programmers.

A database management system (DBMS) is composed of three major parts: (1) a storage subsystem that stores and retrieves data in files; (2) a modeling and manipulation subsystem that provides the means with which to organize the data and to add, delete, maintain, and update the data; (3) and an interface between the DBMS and its users. Several major trends are emerging that enhance the value and usefulness of database management systems:

1. Managers: who require more up-to-data information to make effective decision.

2. Customers: who demand increasingly sophisticated information services and more current information about the status of their orders, invoices, and accounts.

3. Users: who find that they can develop custom applications with database systems in a fraction of the time it takes to use traditional programming languages.

4. Organizations: that discover information has a strategic value; they utilize their database systems to gain an edge over their competitors.

**The Database Model**

A data model describes a way to structure and manipulate the data in a database. The structural part of the model specifies how data should be represented (such as tree, tables, and so on). The manipulative part of the model specifies the operation with which to add, delete, display, maintain, print, search, select, sort and update the data.

**Hierarchical Model**

The first database management systems used a hierarchical model-that is-they arranged records into a tree structure. Some records are root records and all others have unique parent records. The structure of the tree is designed to reflect the order in which the data will be used that is, the record at the root of a tree will be accessed first, then records one level below the root, and so on.

The hierarchical model was developed because hierarchical relationships are commonly found in business applications. As you have known, an organization char often describes a hierarchical relationship: top management is at the highest level, middle management at lower levels, and operational employees at the lowest levels. Note that within a strict hierarchy, each level of management may have many employees or levels of employees beneath it, but each employee has only one manager. Hierarchical data are characterized by this one-to-many relationship among data.

In the hierarchical approach, each relationship must be explicitly defined when the database is created. Each record in a hierarchical database can contain only one key field and only one relationship is allowed between any two fields. This can create a problem because data do not always conform to such a strict hierarchy.

**Relational Model**

A major breakthrough in database research occurred in 1970 when E. F. Codd proposed a fundamentally different approach to database management called relational model, which uses a table as its data structure.

The relational database is the most widely used database structure. Data is organized into related tables. Each table is made up of rows called and columns called fields. Each record contains fields of data about some specific item. For example, in a table containing information on employees, a record would contain fields of data such as a person’s last name, first name, and street address.

Structured query language (SQL) is a query language for manipulating data in a relational database. It is nonprocedural or declarative, in which the user need only specify an English-like description that specifies the operation and the described record or combination of records. A query optimizer translates the description into a procedure to perform the database manipulation.

**Network Model**

The network model creates relationships among data through a linked-list structure in which subordinate records can be linked to more than one parent record. This approach combines records with links, which are called pointers. The pointers are addresses that indicate the location of a record. With the network approach, a subordinate record can be linked to a key record and at the same time itself be a key record linked to other sets of subordinate records. The network mode historically has had a performance advantage over other database models. Today, such performance characteristics are only important in high-volume, high-speed transaction processing such as automatic teller machine networks or airline reservation system.

Both hierarchical and network databases are application specific. If a new application is developed, maintaining the consistency of databases in different applications can be very difficult. For example, suppose a new pension application is developed. The data are the same, but a new database must be created.

**Object Model**

The newest approach to database management uses an object model, in which records are represented by entities called objects that can both store data and provide methods or procedures to perform specific tasks.

The query language used for the object model is the same object-oriented programming language used to develop the database application. This can create problems because there is no simple, uniform query language such as SQL. The object model is relatively new, and only a few examples of object-oriented database exist. It has attracted attention because developers who choose an object-oriented programming language want a database based on an object-oriented model.

**Distributed Database**

Similarly, a distributed database is one in which different parts of the database reside on physically separated computers. One goal of distributed databases is the access of information without regard to where the data might be stored. Keeping in mind that once the users and their data being separated, the communication and networking concepts come into play.

Distributed databases require software that resides partially in the larger computer. This software bridges the gap between personal and large computers and resolves the problems of incompatible data formats. Ideally, it would make the mainframe databases appear to be large libraries of information, with most of the processing accomplished on the personal computer.

A drawback to some distributed systems is that they are often based on what is called a mainframe-entire model, in which the larger host computer is seen as the master and the terminal or personal computer is seen as a slave. There are some advantages to this approach. With databases under centralized control, many of the problems of data integrity that we mentioned earlier are solved. But today’s personal computers, departmental computers, and distributed processing require computers and their applications to communicate with each other on a more equal or peer-to-peer basis. In a database, the client/server model provides the framework for distributing databases.

One way to take advantage of many connected computers running database applications is to distribute the application into cooperating parts that are independent of one anther. A client is an end user or computer program that requests resources across a network. A server is a computer running software that fulfills those requests across a network. When the resources are data in a database, the client/server model provides the framework for distributing database.

A file server is software that provides access to files across a network. A dedicated file server is a single computer dedicated to being a file server. This is useful, for example, if the files are large and require fast access. In such cases, a minicomputer or mainframe would be used as a file server. A distributed file server spreads the files around on individual computers instead of placing them on one dedicated computer.

Advantages of the latter server include the ability to store and retrieve files on other computers and the elimination of duplicate files on each computer. A major disadvantage, however, is that individual read/write requests are being moved across the network and problems can arise when updating files. Suppose a user requests a record from a file and changes it while another user requests the same record and changes it too. The solution to this problem called record locking, which means that the first request makes others requests wait until the first request is satisfied. Other users may be able to read the record, but they will not be able to change it.

A database server is software that services requests to a database across a network. For example, suppose a user types in a query for data on his or her personal computer. If the application is designed with the client/server model in mind, the query language part on the personal computer simple sends the query across the network to the database server and requests to be notified when the data are found.

Examples of distributed database systems can be found in the engineering world. Sun’s Network Filing System (NFS), for example, is used in computer-aided engineering applications to distribute data among the hard disks in a network of Sun workstation.

**英文翻译：**

**数据库管理系统的介绍**

数据库（database,有时被拼作data base）又称为电子数据库，是专门组织起来的一组数据或信息，其目的是为了便于计算机快速查询及检索。数据库的结构是专门设计的，在各种数据处理操作命令的支持下，可以简化数据的存储、检索、修改和删除。数据库可以存储在磁盘、磁带、光盘或其他辅助存储设备上。

数据库由一个或一套文件组成。其中的信息可以分解为记录，每一条记录又包含一个或多个字段（或称为域）。字段是数据存取的基本单位。数据库用于描述实体，其中的一个字段通常表示与实体的某一属性相关的信息。通过关键字以及各种分类（排序）命令，用户可以对多条记录的字段进行查询，重新整理，分组或选择，以实体对某一类数据的检索，也可以生成报表。

所有数据库（除最简单的）中都有复杂的数据关系及其链接。处理与创建，访问以及维护数据库记录有关的复杂任务的系统软件包叫做数据库管理系统（DBMS）。DBMS软件包中的程序在数据库与其用户间建立接口。（这些用户可以是应用程序员，管理员及其他需要信息的人员和各种操作系统程序）

DBMS可组织、处理和表示从数据库中选出的数据元。该功能使决策者能搜索、探查和查询数据库的内容，从而对正规报告中没有的，不再出现的且无法预料的问题做出回答。这些问题最初可能是模糊的并且（或者）是定义不恰当的，但是人们可以浏览数据库直到获得所需的信息。简言之，DBMS将“管理”存储的数据项并从公共数据库中汇集所需的数据项用以回答非程序员的询问。

DBMS由3个主要部分组成：（1）存储子系统，用来存储和检索文件中的数据；（2）建模和操作子系统，提供组织数据以及添加、删除、维护、更新数据的方法；（3）用户和DBMS之间的接口。在提高数据库管理系统的价值和有效性方面正在展现以下一些重要发展趋势：

1.管理人员需要最新的信息以做出有效的决策。

2.客户需要越来越复杂的信息服务以及更多的有关其订单，发票和账号的当前信息。

3.用户发现他们可以使用传统的程序设计语言，在很短的一段时间内用数据库系统开发客户应用程序。

4.商业公司发现了信息的战略价值，他们利用数据库系统领先于竞争对手。

**数据库模型**

数据库模型描述了在数据库中结构化和操纵数据的方法，模型的结构部分规定了数据如何被描述（例如树，表等）：模型的操纵部分规定了数据添加、删除、显示、维护、打印、查找、选择、排序和更新等操作。

**分层模型**

第一个数据库管理系统使用的是分层模型，也就是说，将数据记录排列成树形结构。一些记录根目录，在其他所有记录中都有独立的父记录。树形结构的设计反映了数据被使用的顺序，也就是首先访问处于树根位置的记录，接下来是根下面的记录等。

分层模型的开发是因为分层关系在商业应用中普遍存在。众所周知，一个组织结构图表就描述了一种分层关系：高层管理人员在高层，中层管理人员在较低的层次，负责具体事务的雇员在底层。值得注意的是，在一个严格的分层结构体系中，在每个管理层下可能有多个雇员或多个层次的雇员，但每个雇员只有一个管理者。分层结构数据的典型特征是数据之间的一对多关系。

在分层方法中，当数据库建立时，每一关系即被明确地定义。在分层数据库中的每一记录只能包含一个关键字段，任意两个字段之间只能有一种关系。由于数据并不总是遵循这种严格的分层关系，所以这样可能会出现一些问题。

**关系模型**

在1970年，数据库研究取得了重大突破。E.F.Codd提出了一种截然不同的数据库管理方法，使用表作为数据结构,称之为关系模型.

关系数据库是使用最广的数据结构,数据被组织成关系表,每个表由称作记录的行和称作字段的列组成。每个记录包含了专用项目的字段值。例如，在一个包含雇员信息的表中，一个记录包含了像一个人姓名和地址这样的字段的值。

结构化查询语言（SQL）是一种在关系型数据库中用于处理数据的查询语言。它是非过程化语言或者说是描述性的，用户只须指定一种类似于英语的描述，用来确定操作、记录或描述记录组合。查询优化器将这种描述翻译为过程执行数据库操作。

**网状模型**

网状模型在数据之间通过链接表结构创建关系，子记录可以链接到多个父记录。这种将记录和链接捆绑到一起的方法叫做指针，它是指向一个记录存储位置的存储地址。使用网状方法，一个子记录可以链接到一个关键记录，同时，它本身也可以作为一个关键记录链接到其他一系列子记录。在早期，网状模型比其他模型更有性能优势，但是在今天，这种优势的特点只有在自动柜员机网络，航空预定系统等大容量和高速处理过程中才是最重要的。

分层和网状数据库都是专用程序，如果开发一个新的应用程序，那么在不同的应用程序中保持数据库的一致性是非常困难的。例如开发一个退休金程序，需要访问雇员数据，这一数据同时也被工资单程序访问。虽然数据是相同的，但是也必须建立新的数据库。

**对象模型**

最新的数据库管理方法是使用对象模型，记录由被称作对象的实体来描述，可以在对象中存储数据，同时提供方法或程序执行特定的任务。

对象模型使用的查询语言与开发数据库程序所使用的面向对象的程序设计语言是相同的，因为没有像SQL这样简单统一的查询语言，所以会产生一些问题。对象模型相对较新，仅有少数几个面向对象的数据库实例。它引起了人们的关注，因为选择面向对象程序设计语言的开发人员希望有一个基于在对象模型基础上的数据库。

**分布式数据库**

类似的，分布式数据库指的是数据库的各个部分分别存储在物理上相互分开的计算机上。分布式数据库的一个目的是访问数据信息时不必考虑其他位置。注意，一旦用户和数据分开，通信和网络则开始扮演重要角色。

分布式数据库需要部分常驻于大型主机上的软件，这些软件在大型机和个人计算机之间建立桥梁，并解决数据格式不兼容的问题。在理想情况下，大型主机上的数据库看起来像是一个大的信息仓库，而大部分处理则在个人计算机上完成。

分布式数据库系统的一个缺点是它们常以主机中心模型为基础，在这种模型中，大型主机看起来好像是雇主，而终端和个人计算机看起来好像是奴隶。但是这种方法也有许多优点：由于数据库的集中控制，前面提到的数据完整性和安全性的问题就迎刃而解。当今的个人计算机，部门级计算机和分布式处理都需要计算机之间以及应用程序之间在相等或对等的基础上相互通信，在数据库中客户机/服务器模型为分布式数据库提供了框架结构。

利用相互连接的在计算机上运行的数据库应用程序的一种方法是将程序分解为相互独立的部分。客户端是一个最终用户或通过网络申请资源的计算机程序，服务器是一个运行着的计算机软件，存储着那些通过网络传输的申请。当申请的资源是数据库中的数据时，客户机/服务器模型则为分布式数据库提供了框架结构。

文件服务器指的是一个通过网络提供文件访问的软件，专门的文件服务器是一台被指定为文件服务器的计算机，这是非常有用的。例如，如果文件比较大而且需要快速访问，在这种情况下，一台微型计算机或大型主机将被用作文件服务器。分布式文件服务器将文件分散到不同的计算机上，而不是将它们集中存放到专门的文件服务器上。

后一种文件服务器拥有在其他计算机上存储和检索文件的能力，并可以在每一台计算机上消除重复文件。然而，一个重要的缺点是每个读写请求需要在网络上传播，在刷新文件时可能出现问题。假设一个用户申请文件中的一个数据并修改它，同时另外一个用户也申请这个数据并修改它，解决这种问题的方法叫做数据锁定，即第一个申请使其他申请处于等待状态，直到完成第一个申请，其他用户可以读取这个数据，但不能修改。

数据库服务器是一个通过网络为数据库申请提供服务的软件。例如，假设某个用户在他的个人计算机上输入了一个数据查询命令，如果应用程序按照客户机/服务器模型设计，那么个人计算机上的查询语言通过网络传送到数据库服务器上，当发现数据时发出通知。

在工程界也有许多分布式数据库的例子，如SUN公司的网络文件系统（NFS）被应用到计算机辅助工程应用程序中，将数据分散到由SUN工作站组成的网络上的不同硬盘之间。