

怎么样？被这么一问，是不是发现有一些问题无法简单地解释清楚呢？下面，笔者就公布答案并解释。

答案

初级问题：表（Table）就是被整理成表格形式的数据。

中级问题：DBMS 是 Database Management System（数据库管理系统）的简称。

高级问题：键用于设定表和表之间的关系（Relationship），而索引是提升数据检索速度的机制。

解释

初级问题：一张表由若干个列和行构成。列也被称为字段（Field），行也被称为记录（Record）。

中级问题：市面上的 DBMS 有 SQL Server、Oracle、DB2 等。无论是哪种 DBMS 都可以用基本相同的 SQL 语句操作。

高级问题：其上每个值都能够唯一标识一条记录的字段称为主键。为了在表和表之间建立关系而在表中添加的、其他表主键的字段称为外键。而索引是与键无关的机制。

本章重点

前面的章节讲解的是计算机的构造和程序设计。而本章一改之前的主题，来讲一讲数据库。像 DBMS、关系型数据库、SQL (Structured Query Language, 结构化查询语言)、事务 (Transaction) 之类的数据库术语，想必诸位都有所耳闻吧。可是应该也有很多人觉得自己好像是明白了这些术语的意思，实际上却并没有真正地理解。不仅是数据库，其他计算机技术也一样，不实际地应用，就不能充分掌握。

本章首先介绍数据库的概况，然后通过文字的描述，请诸位体验一下编写简单的数据库应用程序的过程。这样就不但能理解数据库术语的含义，而且还能灵活应用这些知识了。还有一点请诸位明白，在编写数据库应用程序时，可以采用各种各样的方法，而本章所介绍的方法仅仅是其中的一种。

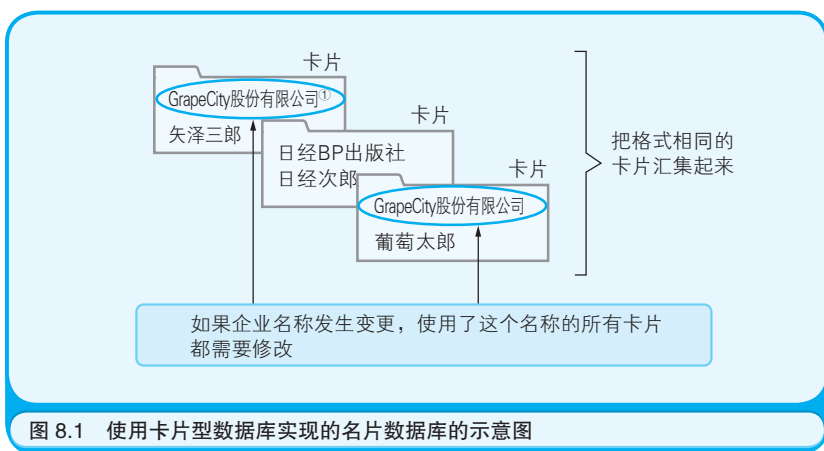


8.1 数据库是数据的基地

所谓数据库 (Database) 就是数据 (Data) 的基地 (Base)。在实施企业的商业战略时，如果企业内的数据散布在各个地方，在更新和检索时就要花费大量时间，分析起来就会很麻烦。但是只要把企业内的数据预先汇集到一个“基地”中并加以整理，各个部门中充满干劲的员工就可以根据需要灵活地使用这些数据。这个数据的基地就是数据库。虽然使用纸质文件整理出来的数据也可以称为数据库，但是利用善于处理数据的计算机整理会更加方便。因为计算机是提高手工工作效率的工具，所以就成为了数据的基地。

把数据存储到计算机中以后，为了将其整理得易于使用，就不得

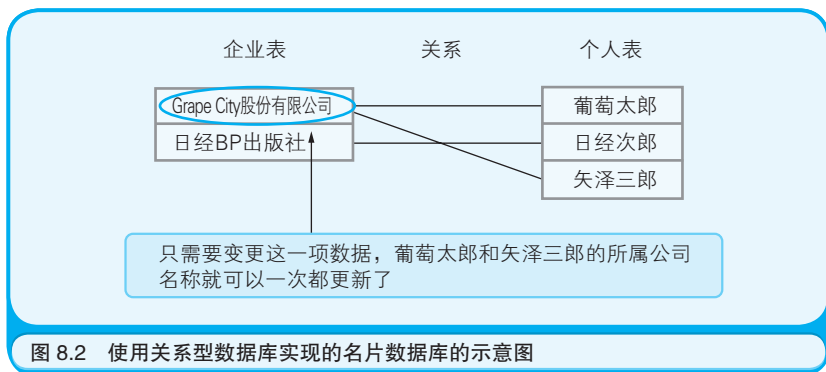
不考虑其存储方式。在手工作业的阶段，通常是像账单或名片那样，把所需的信息汇集到一张纸上。将这样的数据存储形式原封不动地移植到计算机中，就形成了“卡片型数据库”。存储一条数据就好比把一张账单或是名片上的信息记录到一个文件中。卡片型数据库适用于想要实现小规模的数据库的情况。像是地址簿管理程序、存储 Web 电子公告板上的评论等，使用的都是卡片型数据库（如图 8.1 所示）。



可是，如果能够实现能够管理企业所有信息的大规模数据库，卡片型数据库就无能为力了。这是因为卡片与卡片之间缺乏关联性，因此也就难以记录像是“A公司向B公司出售了商品”这样的信息。诸位看了图 8.1 后就会明白，假设公司名称由“GrapeCity 股份有限公司”变更为“葡萄城股份有限公司”，那么麻烦的工作就来了，所有记录了“GrapeCity 股份有限公司”的卡片都需要修改。

① GrapeCity（葡萄城）是一家软件研发公司，总部位于日本仙台，另外在中国、美国、印度、蒙古都设有分支机构。中国葡萄城公司的网址为 <http://cn.grapecity.com/>。——译者注

适合存储大规模数据的是关系型数据库 (Relational Database)。在关系型数据库中, 数据被拆分整理到多张表中, 同时表与表之间的关系也可以被记录下来。对于上面的例子, 只要把数据分别存储到企业表和个人表中, 再在这两张表间建立关系, 那么在公司名称变更时, 只需要更新企业表中的一项数据就能解决问题了, 即把企业表中的“GrapeCity 股份有限公司”改为“葡萄软件股份公司”即可 (如图 8.2 所示)。同时也就能够很方便地记录像是“A 公司向 B 公司出售了商品”这样的数据了。



1970 年美国 IBM 公司的 Codd 先生设计发明了关系型数据库。现在关系型数据库被广泛应用, 以至于一提到数据库就默认是关系型数据库。在后面的章节中, 将要请诸位通过文字上的描述感受其编写过程的数据库应用程序, 也是使用关系型数据库完成的。

8.2 数据文件、DBMS 和数据库应用程序

为了编写数据库应用程序 (即为了便于操作数据库而编写的程序), 诸位可以从零开始埋头编写所有代码, 但是一般情况下, 还是会借助

称作 DBMS 的软件。Microsoft Access、Oracle、SQL Server、DB2 等诸位都有所耳闻吧，这些都是 DBMS 的实例。数据库的实质虽然是某种数据文件，但是诸位编写的应用程序并不是直接去读写这些数据文件，而是以 DBMS 作为中介间接地读写（如图 8.3 所示）。DBMS 不但可以使应用程序轻松地读写数据文件，而且还具有一致并且安全地存储数据的功能。

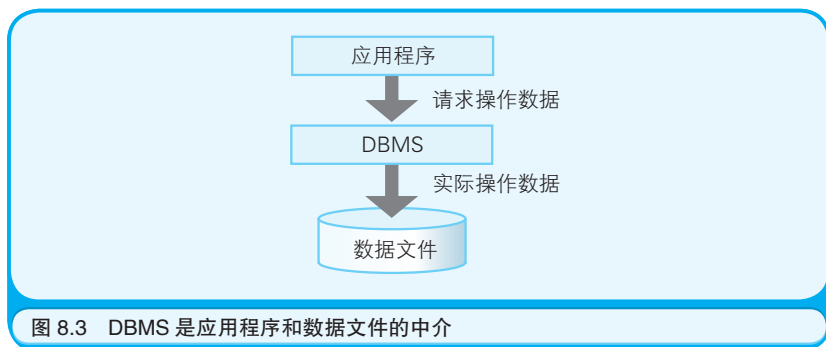


图 8.3 DBMS 是应用程序和数据文件的中介

何为“一致并且安全地存储”将会在后面解释，在此还是先介绍一下数据库系统的构成要素吧。数据库系统的构成要素包括“数据文件”“DBMS”“应用程序”三部分。在小型系统中，把三个要素全部部署在一台计算机上，称作“独立型系统”。在中型系统中，把数据文件部署在一台计算机上，并且使数据文件被部署了 DBMS 和应用程序的多台计算机共享，这样的系统被称为“文件共享型系统”。在大型系统中，把数据文件和 DBMS 部署在一台（或者多台）计算机上，然后用户从另外一些部署着应用程序的计算机上访问，这样的系统被称作“客户端/服务器型系统”。其中部署着数据文件和 DBMS 的计算机是服务器（Server），即服务的提供者；部署着应用程序的计算机是客户端（Client），即服务的使用者。如果把服务器和客户端之间用互

联网联结起来，就形成了 Web 系统。在 Web 系统中，一般情况下应用程序也是部署在服务器中的，在客户端只部署 Web 浏览器（如图 8.4 所示）。

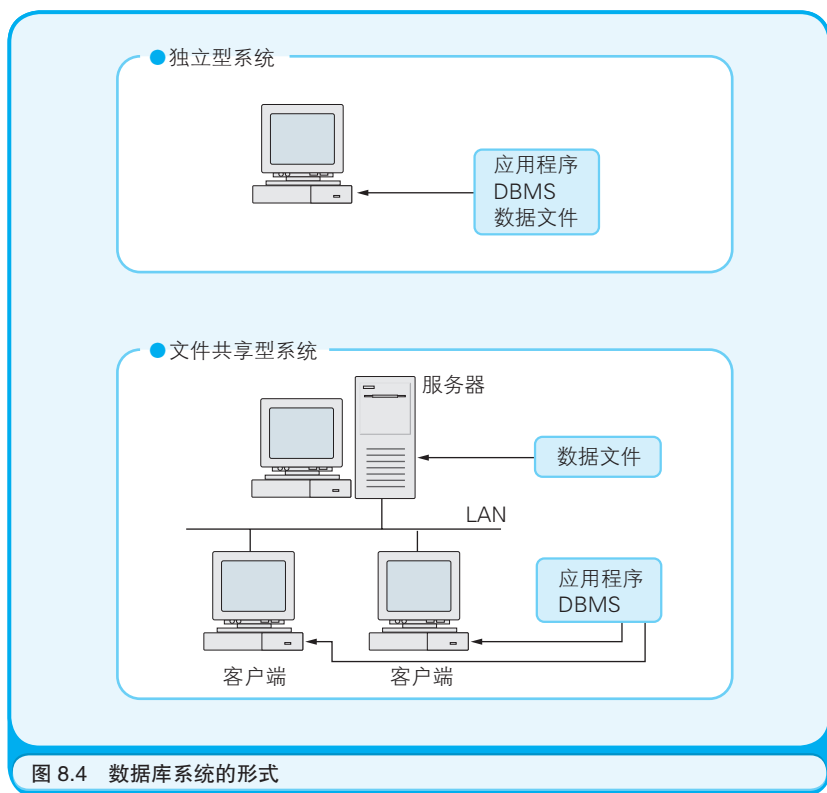


图 8.4 数据库系统的形式

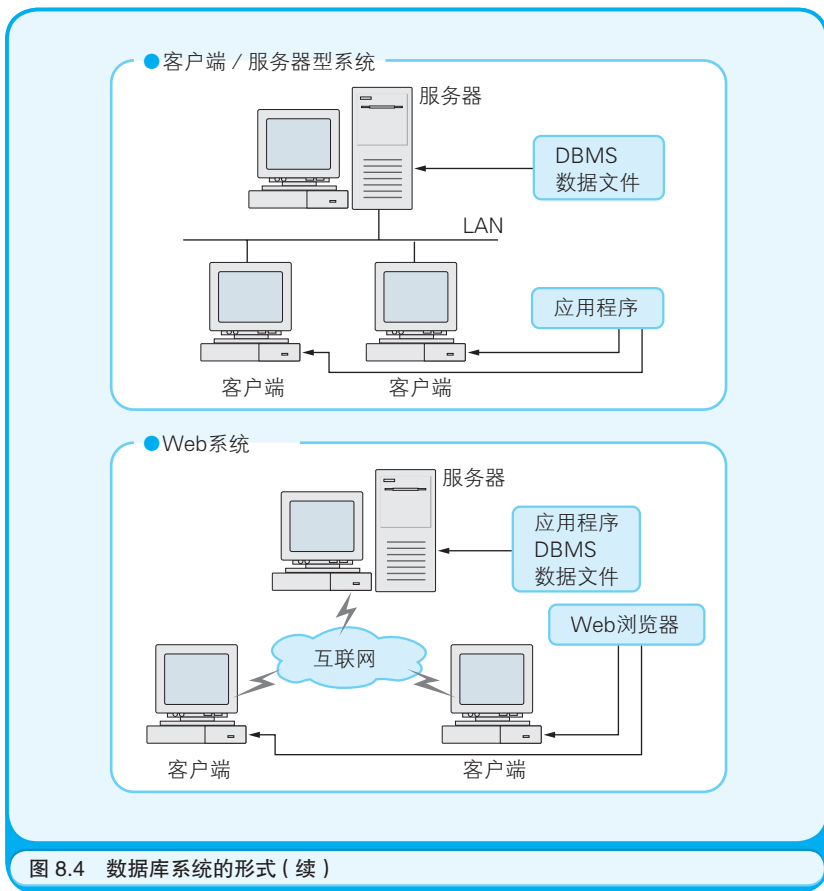


图 8.4 数据库系统的形式 (续)

8.3 设计数据库

既然已经大体上了解了数据库的概况，那么我们就开始实际编写一个数据库应用程序吧。本节中，我们将在一台个人计算机上使用名为 Microsoft Access 的 DBMS 实现一个独立型系统。应用程序部分，使用 Visual Basic 6.0 编写。应用程序以酒铺管理为主题。请诸位学会利用身边的例子来帮助理解新知识。

首先从设计数据库开始。而设计数据库的第一步是从“你想要了解什么”的视角出发找出需要的数据。如果是自己使用的数据库，那么就问问你自己想要了解什么。如果是为客户设计数据库，就要去询问对方想要了解什么。

在酒铺管理的应用程序中，将下面的数据视为客户想要了解的数据。

酒铺经营者需要知道什么？

- 商品名称
- 单价（日元）
- 销售量
- 顾客姓名
- 住址
- 电话号码

当然，仅仅存储这些数据是否够用，是由数据库的使用者决定的。如果缺少了所需的数据，就算使用了数据库，也不能使其发挥作用。反过来，如果包含了不必要的数据，存储包含着这些数据的文件就会白白浪费掉磁盘空间。

把必要的数据筛选出来以后，下一步要考虑的是各种数据的属性。属性也称作模式（内模式），具体来说就是数据的类型（是数字还是字符串），数字的话是整数还是浮点小数，字符串的话最多允许包含多少个字符，是否允许 NULL 值（表示未知或者不存在的值），等等。

几乎所有的 DBMS 都提供了通过可视化界面设置数据属性的工具。通过这种工具，既可以生成逻辑上的表，又可以生成物理上存储数据

的数据文件。其中，表可以被赋予任意的名称。对酒铺经营者所需的各个数据分别设置完属性后，我们将表暂且命名为酒铺表（如图 8.5 所示）。

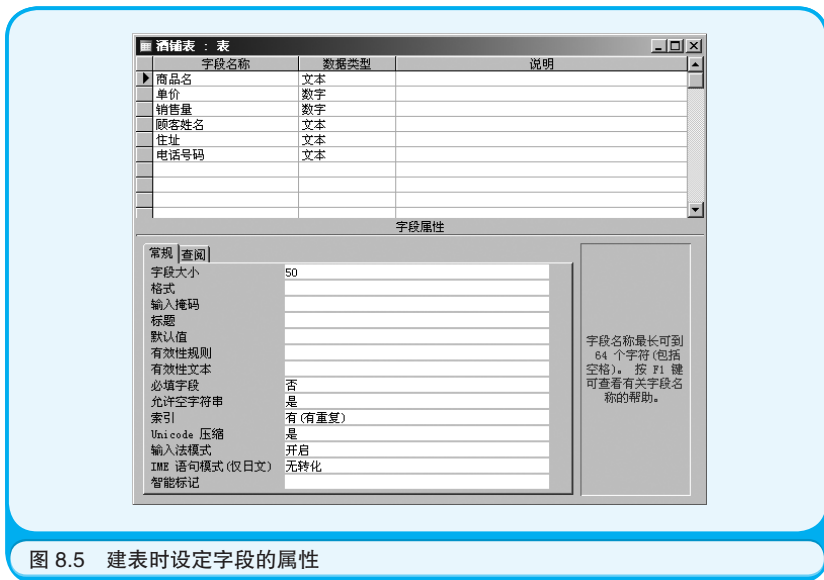


图 8.5 建表时设定字段的属性

在这里，请先记住一些数据库术语。在关系型数据库中，把录入到表中的每一行数据都称为记录，把构成一条记录中的各个数据项（在本例中是商品名称、单价等）所在的列都称作字段。记录有时也被称为行或元组（Tuple），字段有时也被称为列或属性（Attribute）。上面提到的属性（数据的类型）就是设置在字段上的。为了代表字段所存储数据的内容还要为每个字段起一个名字。如图 8.5 所示，通过这个界面定义了构成一条记录的多个字段。之后只要在这个表中录入数据，表就可以使用了。

8.4 通过拆表和整理数据实现规范化

既然表已经准备好了，那么只需要把带有用户界面并且能够读写数据的应用程序做出来，就大功告成了。可实际上却并非如此，如果就这样使用这张表，那么在数据库的运行过程中有可能会产生一些问题。DBMS 既然已经提供了用于手工输入数据的工具，那么我们就先试着录入几条测试数据看看吧（如图 8.6 所示）。



于是这就产生了两个问题。第一个问题是，用户不得不多次录入相同的数据，就像第一条和第二条记录中的数据“日经次郎、东京都千代田区、03-2222-2222”。录入重复数据不仅使应用程序的操作变得繁琐，更白白浪费了磁盘空间。另一个问题是，录入的名称不同指代的却是相同的商品，就像在第三条记录中，应该输入“威士忌”，却错误地输入了“维士忌”，如果让计算机来处理，这种情况就会被识别成不同的商品。也就是说，如果仅使用一张表，就会和应用卡片型数据库（每条记录对应一张卡片）时面临相同的问题。

① 即威士忌一词的错误写法。——译者注