

美 George Tsu-der chou
With W Edward Tiley

HOPE

怎样使用最新数据库 dBASE IV

白为民 唐培顺 译
杨鸿魁 张 继
白为民 校



北京希望电脑公司

George Tsu-der chou
with W. Edward Tiley

怎样使用最新数据库

dBASEIV

北京希望电脑公司

一九九一年十一月

目 录

第一部分 引 言

第一章 dBASEIV和数据库简介

§ 1.1 什么是数据库管理系统	(5)
§ 1.2 关系数据库	(6)
§ 1.3 关系数据库文件的成分	(10)
§ 1.4 关系数据库函数	(12)
§ 1.5 dBASEIV入门	(12)
§ 1.6 小结	(17)

第二章 开始工作

§ 2.1 组织dBASEIV磁盘文件	(18)
§ 2.2 装入dBASEIV程序	(22)
§ 2.3 启动dBASEIV程序	(22)
§ 2.4 dBASEIV速成	(36)
§ 2.5 小结	(37)

第三章 建立和输入数据

§ 3.1 什么是文件目录	(38)
§ 3.2 建立新的文件目录	(39)
§ 3.3 删除已有文件目录	(43)
§ 3.4 建立数据库文件	(45)
§ 3.5 定制数据输入格式	(59)
§ 3.6 使用数据输入格式	(72)
§ 3.7 本章速成	(74)
§ 3.8 小结	(78)

第四章 显示数据

§ 4.1 显示文件目录	(79)
§ 4.2 显示和打印数据库结构	(84)
§ 4.3 显示和打印所有数据记录	(85)
§ 4.4 显示选择的记录和字段	(89)
§ 4.5 本章速成	(96)
§ 4.6 小结	(98)

第五章 编辑数据

§ 5.1 修改已有的数据库结构	(99)
§ 5.2 修改已有的查询设计	(102)
§ 5.3 编辑数据记录	(111)

§ 5.4 本章速成	(120)
§ 5.5 小结	(126)
第六章 对数据进行分类、索引和加总	
§ 6.1 在数据库文件中组织数据	(127)
§ 6.2 分类与索引	(128)
§ 6.3 对数据记录进行分类	(128)
§ 6.4 建立数据库文件的索引	(137)
§ 6.5 在建立索引的数据库文件中查找数据	(152)
§ 6.6 对数据记录进行记数和加总	(154)
§ 6.7 本章速成	(161)
§ 6.8 小结	(169)
第七章 内存变量、表达式和函数	
§ 7.1 内存变量	(171)
§ 7.2 表达式	(177)
§ 7.3 函数	(182)
§ 7.4 小结	(197)
第八章 生成标签和报告	
§ 8.1 在控制中心下创建标签	(198)
§ 8.2 在点提示符下创建标签	(212)
§ 8.3 在控制中心下创建报告	(214)
§ 8.4 在点提示符下创建报告	(235)
§ 8.5 本章速成	(238)
§ 8.6 小结	(247)

第二部分 命令文件程序设计

第九章 命令文件程序设计基础

§ 9.1 定义程序	(248)
§ 9.2 创建并执行dBASE IV 程序	(250)
§ 9.3 控制程序的输出	(255)
§ 9.4 小结	(256)

第十章 输入和输出操作的程序设计

§ 10.1 定义内存变量	(257)
§ 10.2 输入数据到内存变量中	(260)
§ 10.3 输入数据到数据文件中	(264)
§ 10.4 控制屏幕显示	(269)
§ 10.5 显示输出及打印输出	(271)
§ 10.6 小结	(297)

第十一章 条件分支及程序循环

§ 11.1 条件分支	(280)
-------------	-------

§ 11.2 使用程序循环.....	(293)
§ 11.3 小结.....	(298)

第三部分 高级dBASEIV程序设计

第十二章 模块化程序设计

§ 12.1 设计程序模块.....	(300)
§ 12.2 控制程序流程.....	(305)
§ 12.3 在程序模块之间传递内存变量.....	(308)
§ 12.4 建立你自己的模块库.....	(312)
§ 12.5 在程序模块之间传递数据记录.....	(313)
§ 12.6 访问多个数据库文件.....	(315)
§ 12.7 数据库文件的连接.....	(320)
§ 12.8 使用宏函数.....	(324)
§ 12.9 创建菜单.....	(326)
§ 12.10 小结	(329)

第十三章 一个集成化的数据库系统

§ 13.1 建立一个集成化的数据库管理程序.....	(330)
§ 13.2 设计主菜单.....	(330)
§ 13.3 建立开列帐单的子菜单.....	(336)
§ 13.4 发票的创建和打印.....	(341)
§ 13.5 成本维护子菜单的建立.....	(345)
§ 13.6 库存子菜单的建立.....	(347)
§ 13.7 开发雇员情况跟踪模块.....	(353)
§ 13.8 小结.....	(355)
§ 13.9 程序清单.....	(355)

第十四章 dBASE IV 的结构查询语言/SQL

§ 14.1 访问SQL.....	(370)
§ 14.2 在程序中使用SQL.....	(371)
§ 14.3 定义SQL目标.....	(371)
§ 14.4 数据操作.....	(375)
§ 14.5 从SQL表中选取数据.....	(376)
§ 14.6 SQL与dBASE IV 程序的结合.....	(379)
§ 14.7 把dBASE IV 数据传送到SQL表中.....	(380)
§ 14.8 小结.....	(381)

第四部分 命令和函数汇总

dBASE IV 命令汇总

符号约定	(382)
术语定义	(382)

按功能分类的dBASE IV 命令	(383)
命令小结	(389)
dBASE IV函数汇总	
按用途分类的内部函数	(432)
内部函数小结	(435)
附录A ASC II 字符集	(455)
附录B dBASE II plus与dBASE IV的主要差别	(457)
附录C 功能键与控制键小结	(462)
附录D 在硬盘上安装dBASE IV	(464)

第一部分

引言

欢迎您进入dBASEⅣ这个奇妙的世界。就在几年前，只有昂贵的大型计算机系统的用户能够享用数据库管理功能。dBASEⅢ的引入改变了数据的组织方式，并且使之可以在小型的、但功能很强的微机上进行操作。结果使得个人和小型企业受益于数据库管理程序。

dBASEⅣ从dBASEⅢ开始发展，已经过了多种版本。dBASEⅢ提供了数据库管理功能，但因程序是为前几代微机设计的，存贮容量和计算能力均受到限制，不是用户所满意的。dBASEⅣ由dBASEⅢ发展而来，但与之相比有很大的不同，它试图解决上述问题。

虽然dBASEⅢ的数据存贮能力和计算速度有所提高，而且使用方便，人们还是想要一个更加满意、功能更强的程序。Ashton-Tata有限公司是dBASEⅢ的开发者，进而推出了dBASEⅢplus。该版本不仅提供了新增的数据处理命令，而且提供了一个下拉式辅助菜单，使菜单选择更为简单。

为了保持在数据库软件方面的领先地位，Ashton-Tata公司又推出了dBASEⅣ。该软件即将成为微型计算机方面功能最强，同时用户满意的数据库管理系统。dBASEⅣ除增加了一些新命令外，与早期的dBASE系统相比，做出了很大的改进。

这些改进包括下面几点：

- 一个你可方便地进行大多数dBASEⅣ和DOS操作的多功能控制中心（Control Center）。

- 功能强大的实例查询（query by example）（QBE）操作。

- 多索引标记（index tags），你可设置它以便在数据记录改变时，能自动更新。

- 功能强大的且适应性强的报表、标签和数据表格产生器。

- 结构化查询语言（SQL），该语言可提供dBASEⅣ和其它数据库系统之间有效的连接。

借助这些新的改进和新增加的特性，dBASEⅣ树立了它在数据库管理系统中的领导地位，并再次定义了工业标准。

如何使用dBASEⅣ

作为一个数据库管理的工具，dBASEⅣ具有两种处理方式：交互式和批处理方式。采用交互式，你可通过选择适当的菜单选项，或通过直接从键盘上输入易于使用的象英语一样的命令，来建立和处理数据文件。由于计算机反应极快，你可以监控输入和输出。

另一个即批处理方式，它是dBASE系统提供的最重要功能之一。批处理提供了设计一

个综合菜单驱动数据库管理系统所必需的所有功能及灵活性。依据你的需求和经验，你可以用这两种方式管理自己的数据库。

这本《dBASE IV Handbook》（第3版）的主要目的，是告诉你如何使用dBASE IV命令进行有效的数据库管理。为清楚起见，本手册中的例子做到即简单又明了；为保持各例子之间的连贯性，尽量采用同一个数据库来说明多种不同的命令。使用简单的例子，这样你可更好地理解基本原理。一旦你完全理解了数据库管理的基本原理及dBASE IV命令的使用，你便可很容易地设计你自己的更高级的数据库系统。

本书阅读对象

若你拥有dBASE IV，你就应该使用本书。本书并不打算替代dBASE IV手册，而是作为一本用户指南，不仅仅局限于手册中给出的基本概念。将dBASE IV手册作为列写并描述命令和词汇的字典，而将本书作为告诉你如何组合这些命令和词汇的手册。

初学者会觉得本书特别有用。从容易学习的实例中，你可学会为自己的数据管理需要而设计和建立数据库。通过模仿这些实例，你可直接看到用于完成各种功能的命令。前七年均安排速成（QuickStart）一节，在这一节中给了一些简单练习，可用于复习这一章的内容，并练习dBASE IV程序。当你逐渐掌握了dBASE IV命令和操作时，最好的办法莫过于自己实践。

欲升级到dBASE IV的有经验的用户，本书也有很多用处。本书将教你如何使用新的控制中心并强调dBASE IV新的命令。由于dBASE IV命令是按功能分组的，你可将本书作为一本综合性参考手册。（若你没有升级到dBASE IV，在决定是否改为最新版本之前，你可用本书来学习新的特性。）

若你没有dBASE IV，但考虑购买该程序，本书也正适合于你。本书头几章介绍数据库的概念，告诉你数据库如何适应你的数据管理需要，并对dBASE IV及其功能提出了一个总的看法。接下来的几章例子，显示了dBASE IV所提供的功能强大的应用程序。

dBASE IV plus和dBASE IV所提供的的一个最显著的特性，是它的组网能力。由于在建立某个网络时有很多变化，这个问题的讨论已超出本书范围。但是，使用dBASE IV的原理和操作实际上是一样的。你也可使用本书学习有关dBASE IV联网的基本的和高级的特性。

本书的组织方式

本书分为四个主要部分。第一部分“dBASE IV简介”包括1—8章。该部分涉及执行数据库管理任务的步骤，其中包括建立数据库；输入、编辑和显示记录；排序、索引和归纳数据；以及建立报表和标签。对控制中心菜单选项和点提示符命令也作了解释。

第二部分“命令文件编程”由9—11章组成。这部分介绍了命令文件编程的基本原则，并向你显示了如何产生dBASE IV程序以使数据库管理工作便利。

第三部分“高级dBASE IV编程”占12—14章。这几章讨论了程序的高级特性，包括模块化编程和结构化查询语言（SQL）。你可学到如何设计一个多级、菜单驱动的应用程序。

第四部分“命令小结和函数小结”描述并列出了所有dBASE IV命令和函数。

以下部分将逐章简要介绍本书的内容。

第一章“数据库和dBASE IV简介”讨论了数据库管理的基本概念。该章介绍了最常用

的关系数据库模型及dBASEⅣ的基本特性。

第二章“启动”告诉你如何启动dBASEⅣ程序，以及如何用目录组织磁盘文件。本章中，还将介绍功能强大的控制中心菜单系统。速成一节教你启动程序，退回点提示符，取得帮助，及退出dBASE。

第三章“产生并输入数据”开始讨论用于完成基本数据库管理功能的dBASEⅣ命令和菜单选项。本章将说明定义数据结构和产生数据库文件的步骤，描述用于输入数据的菜单选项和点提示符命令。还介绍如何产生和使用定制数据输入表。

第四章“显示数据”教你如何显示和打印数据库文件的结构和内容。除此之外，本章告诉你如何使用功能强大的实例查询（QBE）操作来选择记录。

第五章“编辑数据”包括如何修改数据库文件的内容。你可学习在控制中心内和点提示符下改变数据库结构和数据记录。在速成一节中，练习将教你改变数据库结构，编辑和增加数据记录，以及修改一个查询设计。

第六章“排索、索引和归纳数据”显示了重新安排数据库文件中数据记录的过程。本章描述了用于排序数据记录和给文件编索引的方法。实例说明了这些操作，并强调了排序对编索引的优势。

第七章“内存变量、表达式和函数”说明了如何在被称为变量的暂时内存位置对数据排序，并显示了在数据处理中表达式的用处。本章还将告诉你如何执行高级数学运算，及如何用dBASEⅣ的内部函数将数据从一种类型转换为另一种类型。

第八章“产生标签和报表”教你如何使用dBASEⅣ的内部标签和报表发生器，来产生标签和报表。实例将向你提供用于设计、编辑和使用定制标签和报表的有效选项。

第九章“命令文件编程的基本规则”介绍批处理方式的使用。本章告诉你如何将dBASEⅣ命令组合成一个程序文件，然后该文件中的指令以批处理方式由计算机执行。通过使用大量的例子，本章展示了批处理方式的强大功能和灵活性。

第十章“编程输入和输出操作”强调可以执行程序文件中输入和输出操作的方式。本章显示了如何使用输入命令键入和修改数据，以及用输出命令将数据输出至屏幕和打印纸上。

第十一章“条件分支和程序循环”提供了使用条件分支和程序循环而改变正常处理顺序的方式。本章包括使用DO WHILE命令和其它命令来完成数据库管理功能，这些功能需要重复性操作。

第十二章“模块化编程”集中于结构化编程的概念。在本章中，你可学到如何将一个数据库管理系统分成几个小的且易于管理的子系统或过程，然后可通过一个多级菜单将它们连为一体。你还可以学到如何设计一个使用若干不同数据库文件的数据库管理程序。

第十三章“集成化数据库系统户提供了一个模型，显示一个完整的、小型商业数据库管理系统。本章集中于多级、菜单驱动式应用程序，该程序包含记帐、控制清单和保持顾客帐单等几个模块。从实例中，你可看到如何在一个菜单驱动式dBASE程序中，设计和集成各种不同的数据管理函数。

第十四章“dBASEⅣ中SQL简介”描述了功能强大的结构化查询语言命令。你将学会使用这些命令进行数据库管理操作的方法，以及将SQL并入你的dBASEⅣ程序中的方法，本章的目标是向你介绍dBASEⅣSQL，使你理解由此而引入的有关概念。

dBASE IV 命令小结提供了dBASE IV 命令的语法和实例。dBASE IV 函数小结列出了dBASE IV 的内部函数。

本书中还包括四个有用的附录。附录A包括标准ASCII 字符码，附录B归纳了dBASE III plus与dBASE IV 之间的区别。附录C列出了功能键和控制键清单，附录D是在硬盘上安装dBASE IV 的指南。（译者注：附录C和附录D的内容原文概括有误，已纠正）

第一章 数据库和dBASEIV简介

数据库管理系统 (DBMS) 早已应用于组织和处理大量商业数据, DBMS的核心是数据库。随着你选择的设计方式不同, 数据库的形式也不相同。由于dBASE IV是一个关系数据库系统, 所以本书仅讨论数据库的关系模型。

本章开头说明数据库的基本设计, 并采用一个数据库实例描述一个典型公司的组织结构, 来说明关系数据库的设计宗旨。之后你将看到dBASE IV如何实现数据库管理系统这样一个应用。在随后的各章中将讨论用dBASE IV建立和使用数据库的操作及步骤, 本章还将提供给你该dBASE IV程序的概貌。

§ 1.1 什么是数据库管理系统

“数据库管理系统” (database management system) 一词指在一个大型计算机系统中大量信息的系统化组织与管理。尽管数据处理人员对该词仅使用了一段时间, 但自从微机dBASE I程序的引入, 它已是众所周知的了。

数据库 (database) 是以某种特定方式组织起来的有用信息集合。例如, 你可以将一个私人电话簿看作一个数据库。

James C. Smith	(206) 123-4567
Albert K. Zeller	(212) 457-9801
Doris A. Gregory	(503) 204-8567
Harry M. Nelson	(315) 576-0235
Tina B. Baker	(415) 787-3154
Kirk D. Chapman	(618) 625-7843
Barry W. Thompson	(213) 432-6782
Charles N. Duff	(206) 456-9873
Winston E. Lee	(503) 365-8512
Thomas T. Hanson	(206) 573-5085

图1.1 一个Telephone数据库

该电话本是一个随机列出的姓名和电话号码组成的表——即按照输入的顺序编排。但你也可以依自己所好以某种顺序或形式编排这些电话号码。例如, 你可以按区域号码 (area code) 并在各区域号码中按姓 (last name) 的字母顺序, 将输入项分组。

Charles N. Duff	(206) 456-9873
Thomas T. Hanson	(206) 573-5085
James C. Smith	(206) 123-4567
Albert K. Zeller	(212) 457-9801
Barry W. Thompson	(213) 432-6782
Harry M. Nelson	(315) 476-0235
Tina B. Baker	(415) 787-3154
Doris A. Gregory	(503) 204-8567
Winston E. Lee	(503) 365-8512
Kirk D. Chapman	(618) 625-7843

图1.2 Telephone数据库中已排序的数据项

除了根据他们各自的姓重排电话号码外, 还可以对表添加新的项目, 修改已有的项目, 或从表中删除不要的项,

电话号码数据库是一个非常简单的例子。各项独立，相互间没有联系。更复杂的数据库中各项间有联系。例如，图1.3所示的一个公司的组织结构，它表示了一个各项相关的数据库。

在图1.3所示的数据库中，有关各雇员及其在公司中的职位信息以数据项存贮并按某种方式排列。此外，这些雇员及职位的关系信息在数据库中也要以数据项存贮。在库中这些数项的组织 and 存贮方式取决于你所选用的设计方式。

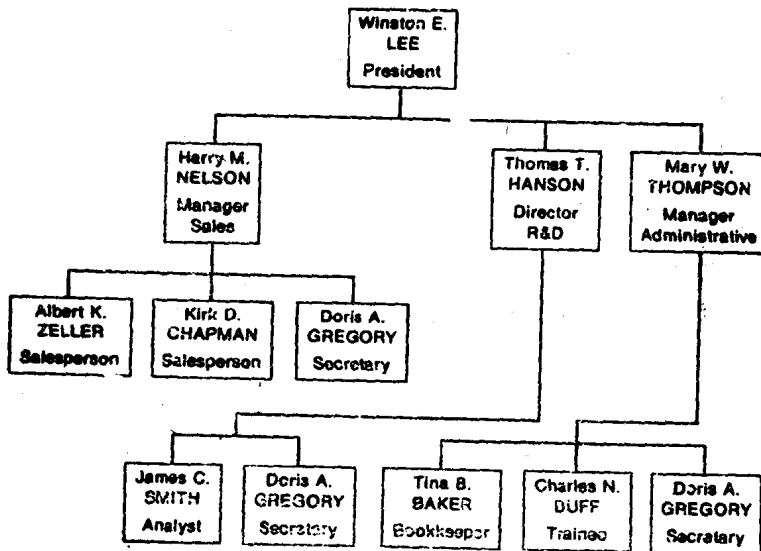


图1.3 一个典型的组织结构

§ 1.2 关系数据库

根据你选择的模型，数据库中的信息可以用多种方式组织。这些数据库模型包括层次、网状和关系数据库。因为dBASE IV是一个关系数据库系统，本书仅限于关系数据库的概念和应用。

§ 1.2.1 数据表

关系模型将数据元素安排在一个或多个二维数据表中，各表由许多行和列组成。每行包含的信息属于数据库中的一项。一行中的数据又被分为几项，每项占表中一列。

例如，上一节中提到的电话簿例子，可以安排为如下一个数据表（见图1.4）。

(4)

Data Table: Phone			
Last Name	First Name	Area Code	Phone Number
Smith	James C.	206	123-4567
Zeller	Albert K.	212	457-9801
Gregory	Doris A.	503	204-8567
Nelson	Harry M.	315	576-0235
Baker	Tina B.	415	787-3154
Chapman	Kirk D.	618	635-7043
Thompson	Mary W.	213	432-6782
Duff	Charles N.	206	456-9073
Lee	Winston E.	501	365-8512
Hanson	Thomas T.	206	573-5085

图1.4中所含信息代表了一个数据表，被命名为phone。数据表的各行包含了某个电话

号码及其拥有者的信息，而各列为描述该行信息的一个值。

一旦将所需数据填入数据表后，你便可通过指定某个筛选条件 (filter condition) 显示表中的选择项。例如，如果需要了解所给区域码 (比如206) 的所有电话号码，你可要求显示在其中一列中具有206区域码的那些行。

§ 1.2.2 单表数据库

一个简单的电话数据库包含如上所示phone例子那样的单个数据表。类似地，图 1.3 的组织结构中的信息也可编排成一个名为雇员 (Employee) 的单个数据表 (见图1.5)。

Data Table: Employee			
Name	Position	Department	Supervisor
Winston E. Lee	President		
Harry M. Nelson	Manager	Sales	Winston E. Lee
Thomas T. Hanson	Director	R&D	Winston E. Lee
Mary W. Thompson	Manager	Admin.	Winston E. Lee
Albert K. Zeller	Salesperson	Sales	Harry M. Nelson
Kirk D. Baker	Salesperson	Sales	Harry M. Nelson
Doris A. Chapman	Secretary	Admin.	Harry M. Nelson
Doris A. Chapman	Secretary	Admin.	Thomas T. Hanson
Doris A. Chapman	Secretary	Admin.	Mary W. Thompson
James C. Smith	Analyst	R&D	Thomas T. Hanson
Tina B. Baker	Bookkeeper	Admin.	Mary W. Thompson
Charles N. Duff	Trainee	Admin.	Mary W. Thompson

图1.5 一个单表数据库

在Employee数据表中，采用了12行、4列来存贮有关公司的组织结构信息。前3列分别描述雇员的姓名 (name)、职位 (position) 和工作部门 (department)，最后一列表明雇员的经理 (supervisor)。

你可能认为将数据库设计为一个单表数据库更简洁和容易，但是单表数据库也存在一些问题。

问题之一是在单表数据中，描述数据项之间的多重关系时需要用多余的数据项填写。例如图1.5，需用三项定义Doris和她的三个经理：Harry、Thomas和Mary的关系。当然，必须定义这些多重关系。但是将Doris的个人信息重复三次是多余的。采用关系数据库可消除这种多余。

在Employee数据表中，大多数行的相应列填有数值。一个例外是第一行“公司经理”，不属于指定部门，没有监管经理。结果第一行的最后两列空着或输入空值。在数据表中应避免空值，因为它们在数据操作中会产生问题，因此不允许在一些数据库系统中使用。

在一个单表数据库中，空值常常是不可避免的。例如，如果一位经理辞职而新的人选未定，所有由该经理管的雇员的监督人一列将为空值。

所以应设计一下数据库，使它可以利用许多数据表存贮信息。例如，一个处理图 1.1 中组织结构信息的数据库系统，使用了三张数据表：一张存贮经理信息，一张存贮其下属信息，还一张描述经理与其下属的关系。存贮经理及其下属信息的数据表可如图1.6所示的构造。

Data Table: Manager			
M_ID	M_Name	M_Position	M_Department
123-45-6789	Winston E. Loe	President	
412-63-3213	Harry M. Nelson	Manager	Sales
205-33-9850	Thomas T. Hanson	Director	R&D
521-45-3423	Mary W. Thompson	Manager	Admin.

Data Table: Staff		
S_ID	S_Name	S_Position
732-07-8219	Albert K. Zeller	Salesperson
356-87-4231	Kirk D. Baker	Salesperson
124-76-0234	Doris A. Chapman	Secretary
572-45-8123	James C. Smith	Analyst
565-34-5570	Tina B. Baker	Bookkeeper
555-23-4572	Charles M. Duff	Trainee

图1.6 经理和下属数据表

经理及其下属之间的关系在数据表中也可以定义为记录。例如，如图1.7所示，建立称为Supervise的数据表来存贮经理及其下属的关系。

Data Table: Supervise	
M_Name	S_Name
Harry M. Nelson	Albert K. Zeller
Harry M. Nelson	Kirk D. Chapman
Harry M. Nelson	Doris A. Gregory
Thomas T. Hanson	James C. Smith
Thomas T. Hanson	Doris A. Gregory
Mary W. Thompson	Tina B. Baker
Mary W. Thompson	Charles M. Duff
Mary W. Thompson	Doris A. Gregory

图1.7 一个关系数据表

总之，Manager和staff数据表可与supervise关系表相结合构成ER图（实体联系图）。ER图表示两个实体之间的关系——在本例中即Manager和Staff的关系（见图1.8）。

在Supervise关系数据表中，各行描述了经理姓名(M_Name)及其下属姓名(S_Name)。从关系数据表中你会发现，经理及其某个下属的姓名在表中出现若干次，而有些下属的姓名仅出现一次。这种情况是由于这些经理及其下属之间存在着不同类型的关系造成的。

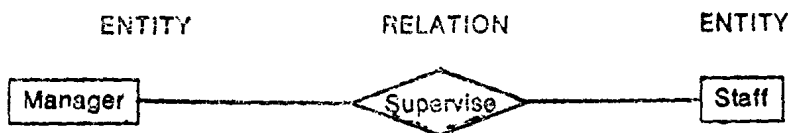


Fig. 1.8

An Entity-Relation (ER) diagram.

图1.8 实体关系(ER)图

§ 1.2.3 关系

关系是两个或两个以上实体之间的连接 (link)、联合 (association) 或联系 (connection)，一个实体可被定义为数据库管理应用中的一个有趣物体。在我们的例子中，公司的经理或下属看作实体，并定义一个关系表示经理及其下属间的监管联系。

两个实体之间基本上存在着三种关系：一对一，多对一，多对多关系。

在一对一关系数据库模型中，第一个实体中的一项仅与第二个实体中的项有关。定义为一对一关系模型的组织结构，要求一名经理只监管一名下属而每位下属只由一名经理监管。

假设公司有6名雇员 (Herry, Thomas, Mary, Albert, James, 和Tina), 其一对一关系数据库模型如图1.9所示。

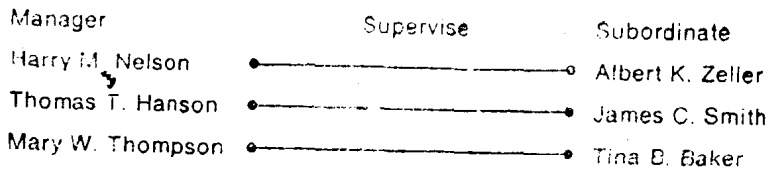


图1.9 一对一关系数据库模型

有了一对一关系模型后, 在Supervise关系数据表中, 每个关系就可定义为一项 (见图1.10)

Data Table: Supervise

M_Name	S_Name
Harry M. Nelson	Albert K. Zeller
Thomas T. Hanson	James C. Smith
Mary W. Thompson	Tina B. Baker

图1.10 描述一对一关系

在多对一关系模型中, 实体中的一项对应于另一实体中的几项 (一项或一项以上)。在组织结构一例中, 即一个经理监管一个或多个下属, 但每个下属只由一个经理监管。同样, 也可建立这样一个组织结构, 即一个下属由一个或多个经理监管; 但每个经理只监管一个下属。

举个例子, 前面在图1.3中所完的组织结构可改为多对一模型的形式 (见图1.11)。

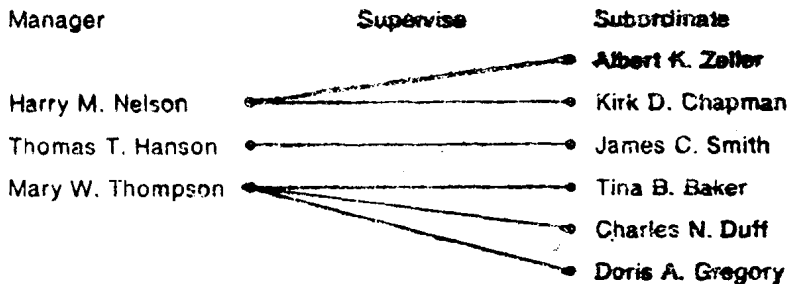


图1.11 多对一关系

从图1.11可见, 经理Harry监管两位下属, Albert和kirk, 而每位下属只由一位经理监管。在Mary和她的三个下属 (Tina, Charles和Doris) 之间也存在类似的关系。在多对一 (或一对多) 关系模型的数据表中, 需要一项以上来描述它们的关系。例如, 在Supervise关系数据表中需要三项表示Mary与其下属的关系 (见图1.12)。

最常见的关系模型是多对多模型, 即一实体中的一项对应另一实体中的一项或多项, 反之亦然。最初在图1.3中定义的组织结构就是这样一个多对多关系模型; 即, 公司的一个或多个经理监管一个或多个下属, 而某些下属由一个或多个经理监管 (见图1.13)。

Data Table: Supervise

M_Name	S_Name
Mary W. Thompson	Tina B. Baker
Mary W. Thompson	Charles N. Duff
Mary W. Thompson	Doris A. Gregory

图1.12 描述多对一关系

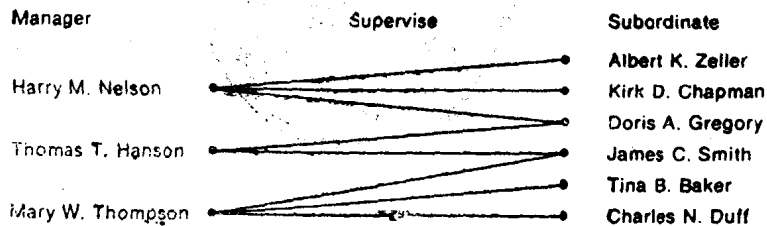


图1.13 多对多关系

如同在多对一关系模型中一样，在数据表中需要多项描述这些多对多关系。定义图1.1₁所描述的关系的项已在Supervise数据表中给出（参见图1.7）。

§ 1.3 关系数据库文件的成分

如前面的例子所示，数据表代表了关系数据库的主干（backbone）。数据表包括了一定数量的行和列，每行称作一个数据记录，每列称为一个数据域。给数据表赋一个名，并作为数据文件保存起来。

虽然数据库可以包含若干个数据表，你也可以建立一个很简单的数据库，象Employee数据表那样，只有一个数据表。这样一个数据表的数据域，在dBASE IV数据库结构中指定，并成为数据库文件的不可分割的部分（见图1.14）。

Layout Organize Append Go To Exit						Bytes remaining: 3958
Num	Field Name	Field Type	Width	Dec	Index	
1	FIRST_NAME	Character	10		N	
2	LAST_NAME	Character	10		V	
3	AREA_CODE	Character	3		V	
4	PHONE_NO	Character	8		N	
5	MALE	Logical	1		N	
6	BIRTH_DATE	Date	8		V	
7	ANNUAL_PAY	Numeric	8	2	N	

Database: C:\databases\FPL\PLUFF.DBF Size: 142
 Enter the field name. Insert/Delete field: Ctrl-W/Ctrl-U
 Field names begin with a letter and may contain letters, digits and underscores

图1.14 一个dBASE IV数据库结构

除了数据库结构外，数据库文件存储含雇员信息的数据记录（见图1.15）。

Records Fields Go To Exit						
FIRST_NAME	LAST_NAME	AREA_CODE	PHONE_NO	MALE	BIRTH_DATE	ANNUAL_PAY
Albert N.	Smith	206	423-4567	T	07/04/58	22000.00
Doris A.	Zeller	212	457-3800	F	09/20/59	27900.00
Harry M.	Gregory	203	284-8567	F	07/04/62	16900.00
Tina B.	Nelson	315	576-0235	T	12/15/58	29000.00
Kirk D.	Baker	415	787-3150	T	11/12/56	25900.00
Mary U.	Chapman	218	625-7845	F	08/04/61	19750.00
Charles M.	Thompson	219	432-6783	F	05/10/55	24500.00
Winston E.	Duff	286	456-9873	T	07/12/64	13500.00
Thomas T.	Lee	503	389-9003	T	05/11/39	34900.00
	Hanson	206	573-8905	T	12/01/45	28950.00

View and edit fields

图1.15 dBASE IV 数据记录列表

§ 1.3.1 数据库结构

数据库结构包含数据记录中各字段的详细描述。这些细节包括下列条目：

Description	Information Entered
Field name:	Name or identification of the data field
Field type:	Kind of data field
Field width:	Dimension of the data field

在dBASE IV中，还可以指明你想要编入索引的数据记录。编索引 (indexing) 是在给定的数据表中整理数据记录的过程，它能以某种方式恢复或显示信息。

定义结构有多种用途。操作数据时，你可以用结构中指定的字段名恢复和查询这些域中存储的数据。例如，可以在整个数据库应用中用LAST NAME字段名找出与某个特定姓有关的信息。

数据类型的说明指出信息的使用方法。如果将数据字段定义为数字型，则字中的项可用于运算式中。但如果字段中的数据定义为字符型，则只能在查寻操作中作为标号或目标 (object)；字符串不能用于运算式中，即使字符串由数字组成。

由于dBASE IV认为数据域长度固定，因此要根据将要输入字段中的最大数据项所具有的字符数，确定数据字段的大小。通过说明数据字段的大小，就可以给数据项保留必要的存储空间。如果数据库中最长的姓为10个字符，则应将LAST NAME域的大小定义为10，即使有些姓只需不到10个字符。

§ 1.3.2 数据记录

如前所述，数据库文件的内容被组织成数据记录和字段。数据记录含有单数据项入口。例如，在Employee数据库文件中，有关雇员的信息构成一个数据记录。

通常将数据库文件中的数据记录按照其输入次序编排。

在各数据记录加入数据库中时，赋给它一个顺序号，此后，用户即可根据记录号识别这