



全国高等教育自学考试指定教材 电子商务专业（高级证书）

互联网数据库

附：互联网数据库自学考试大纲

课程代码
0911
[2006年版]

组编 / 全国高等教育自学考试指导委员会

主编 / 周志忠

本教材附赠网络学习卡

中国财政经济出版社



封面设计/寻木

ISBN 978-7-5005-9352-2



9 787500 593522 >

总定价：53.50元(共两册)

全国高等教育自学考试指定教材
电子商务专业

互联网数据库

(2006年版)

(附：互联网数据库自学考试大纲)

全国高等教育自学考试指导委员会 组编
主 编 周志忠

中国财政经济出版社

图书在版编目 (CIP) 数据

互联网数据库: 附互联网数据库自学考试大纲/周志忠主编. —北京: 中国财政经济出版社, 2006.9

全国高等教育自学考试指定教材

ISBN 978-7-5005-9352-2

I. 互… II. 周… III. 互联网络-关系型数据库-数据库管理系统-高等教育-自学考试-教材 IV. ①TP311.138②TP393.4

中国版本图书馆 CIP 数据核字 (2006) 第 109108 号

中国财政经济出版社 出版

URL: <http://www.cfeph.cn>

E-mail: cfeph@cfeph.cn

(版权所有 翻印必究)

社址: 北京市海淀区阜成路甲 28 号 邮政编码: 100036

北京市荣盛彩色印刷有限公司印刷

787×1092 毫米 16 开 35.25 印张 812 000 字

2006 年 11 月第 1 版 2010 年 10 月北京第 5 次印刷

定价: 53.50 元

ISBN 978-7-5005-9352-2/TP.0134

(所购教材如有印装问题, 请在当地教材供应部门调换)

组 编 前 言

21 世纪是一个变幻难测的世纪，是一个催人奋进的时代。科学技术飞速发展，知识更替日新月异。希望、困惑、机遇、挑战，随时随地都有可能出现在每一个社会成员的生活之中。抓住机遇，寻求发展，迎接挑战，适应变化的制胜法宝就是学习——依靠自己学习，终生学习。

作为我国高等教育组成部分的自学考试，其职责就是在高等教育这个水平上倡导自学、鼓励自学、帮助自学、推动自学，为每一个自学者铺就成才之路。组织编写供读者学习的教材就是履行这个职责的重要环节。毫无疑问，这种教材应当适合自学，应当有利于学习者掌握、了解新知识、新信息，有利于学习者增强创新意识、培养实践能力、形成自学能力，也有利于学习者学以致用，解决实际工作中所遇到的问题。具有如此特点的书，我们虽然沿用了“教材”这个概念，但它与那种仅供教师讲、学生听，教师不讲、学生不懂，以“教”为中心的教科书相比，已经在内容安排、形式体例、行文风格等方面都大不相同了。希望读者对此有所了解，以便从一开始就树立起依靠自己学习的坚定信念，不断探索适合自己的学习方法，充分利用已有的知识基础和实际工作经验，最大限度地发挥自己的潜能，达到学习的目标。

欢迎读者提出意见和建议。

祝每一位读者自学成功。

全国高等教育自学考试指导委员会

2005 年 8 月

编 者 的 话

数据库一直是计算机科学的重要研究方向之一，从最初的数据存储系统发展到如今的关系型数据库和面向对象数据库，在数据容量、查询速度和数据保护等方面均有质的飞跃。在数据库的实际应用飞速发展的同时，数据库理论也日臻完善，形成了整套的关系数据库理论体系，并初步形成了面向对象数据库的理论体系。数据库理论体系的完善反过来又促进了数据库应用水平的提高，容量更大、效率更高、功能更强的数据库系统得以实现。数据库系统已经成为信息时代信息存储的基石，也是当前计算机在各个行业中最普遍的应用，因此，掌握数据库的理论和应用是十分重要的。

互联网技术是近年来发展最快的计算机科学分支，通过互联网，可以迅速通畅地将信息传递到世界上各个角落。互联网技术的发展使计算机这个概念上用来进行科学计算的工具有变成了信息终端，人们坐在计算机前就可以轻松获得所需信息。而当前互联网上的许多应用，比如说网站搜寻、电子邮件、电子布告牌系统等等，其根基都是建立在数据库系统上的，这就形成了一个新的研究领域——互联网数据库。

作为一个交叉学科，互联网数据库技术实际上是网络技术和数据库技术的完美结合。因此本书在安排上同时顾及了对网络基础知识和互联网基础知识的介绍，然后再融合两个领域，系统介绍互联网数据库的概念和应用。本书前六章围绕数据库的基本知识展开，从数据库系统的发展演变、数据库理论基础、SQL语言和数据库保护技术等诸多方面详细介绍了数据库理论体系，在此基础上，第七章引入了对网络的介绍，简单明了地讲述了互联网的基本工作原理和互联网数据库的实现原理，最后从第八章到第十一章介绍了当今流行的互联网数据库技术（如ASP和JDBC）和一些比较典型而且成熟的商用数据库系统，并对互联网数据库的发展方向做出了展望。

本教材力图用最明了最易懂的语言和详尽的实例来解释数据库理论中比较枯燥且抽象的知识，并努力做到互联网知识和数据库知识的有机结合。由于编写时间仓促，笔者水平有限，书中难免有欠妥之处，恳请广大读者批评指正。

编 者

2006年6月

目 录

第一章 绪论	(1)
1.1 数据管理技术的发展	(1)
1.1.1 人工管理阶段	(1)
1.1.2 文件系统阶段	(2)
1.1.3 数据库系统阶段	(3)
1.1.4 数据库技术的产生和发展	(6)
1.2 数据模型	(8)
1.2.1 数据模型的要素	(8)
1.2.2 概念模型	(9)
1.2.3 数据模型	(13)
1.3 数据库系统的结构	(21)
1.3.1 数据库系统模式的概念	(21)
1.3.2 数据库系统的三级模式结构	(21)
1.3.3 数据库的二级映象功能与数据独立性	(23)
1.4 数据库管理系统	(24)
1.4.1 数据库管理系统的功能与组成	(24)
1.4.2 数据库管理系统的工作过程	(26)
第二章 关系数据库简介	(28)
2.1 关系模型的基本概念	(28)
2.1.1 关系数据结构及形式化定义	(29)
2.1.2 关系的完整性	(33)
2.2 关系代数	(36)
2.2.1 传统的集合运算	(37)
2.2.2 专门的关系运算	(37)
2.3 关系演算	(43)
2.3.1 元组关系演算语言 ALPHA	(43)
2.3.2 域关系演算语言 QBE	(47)
第三章 关系数据库标准语言 SQL	(51)
3.1 SQL 概述	(51)
3.1.1 SQL 的特点	(51)

3.1.2	SQL 语言的基本概念	(53)
3.2	数据定义	(54)
3.2.1	定义、删除与修改基本表	(54)
3.2.2	建立与删除索引	(55)
3.3	数据操纵	(56)
3.3.1	SQL 查询语句	(56)
3.3.2	SQL 更新语句	(62)
3.4	视图	(64)
3.4.1	定义视图	(64)
3.4.2	查询视图	(67)
3.4.3	更新视图	(67)
3.4.4	视图的作用	(69)
3.5	数据控制	(70)
3.5.1	授权	(70)
3.5.2	收回权限	(71)
第四章	关系数据库设计理论	(73)
4.1	数据依赖	(73)
4.1.1	关系模式中的数据依赖	(73)
4.1.2	数据依赖对关系模式的影响	(74)
4.1.3	相关概念	(75)
4.2	范式	(76)
4.2.1	第一范式 (1NF)	(77)
4.2.2	第二范式 (2NF)	(79)
4.2.3	第三范式 (3NF)	(80)
4.2.4	BC 范式 (BCNF)	(81)
4.2.5	多值依赖与第四范式 (4NF)	(83)
4.3	关系模式的规范化	(86)
4.3.1	关系模式规范化的步骤	(86)
4.3.2	关系模式的分解	(87)
第五章	数据库保护	(92)
5.1	安全性	(92)
5.2	完整性	(96)
5.2.1	完整性约束条件	(97)
5.2.2	完整性控制	(99)
5.3	并发控制	(101)
5.3.1	基本概念	(102)
5.3.2	封锁 (Locking)	(103)
5.3.3	活锁和死锁	(105)
5.3.4	可串行性	(106)

5.3.5	两段锁协议	(107)
5.4	恢复	(108)
5.4.1	故障的种类	(108)
5.4.2	转储和恢复	(110)
5.4.3	日志文件	(111)
第六章	数据库设计	(113)
6.1	数据库设计概述	(113)
6.1.1	数据库和信息系统	(113)
6.1.2	数据库设计的特点	(114)
6.1.3	数据库设计方法简述	(114)
6.1.4	数据库设计的基本步骤	(115)
6.2	需求分析	(117)
6.2.1	需求分析的任务	(118)
6.2.2	需求分析的方法	(118)
6.2.3	数据字典	(119)
6.3	概念结构设计	(121)
6.3.1	概念结构	(121)
6.3.2	概念结构设计的方法与步骤	(122)
6.3.3	数据抽象与局部视图设计	(122)
6.3.4	视图的集成	(126)
6.4	逻辑结构设计	(130)
6.4.1	E-R图向数据模型的转换	(131)
6.4.2	数据模型的优化	(133)
6.4.3	设计用户子模式	(134)
6.5	数据库物理设计	(135)
6.6	数据库实施与维护	(137)
第七章	基于 Web 数据库技术概述	(141)
7.1	WWW 技术——超文本传输协议 (HTTP)	(141)
7.2	公共网关接口 (CGI)	(143)
7.2.1	CGI 的一般概念	(143)
7.2.2	CGI 程序的工作方式	(144)
7.2.3	CGI 的特点	(146)
7.3	服务器 API	(146)
7.3.1	服务器 API 概述	(147)
7.3.2	ISAPI	(147)
7.3.3	Internet Database Connector	(149)
7.4	Java 数据库连接	(150)
7.4.1	JDBC	(151)
7.4.2	DAO	(151)

7.5	Microsoft ActiveX 平台	(151)
第八章	JDBC——基于 Java 的数据库连接	(157)
8.1	JDBC 概述	(157)
8.2	JDBC 的设计目标	(161)
8.2.1	分布式模块	(162)
8.2.2	JDBC 的远见	(163)
8.3	主要的 JDBC 类	(163)
8.3.1	JDBC 驱动器	(166)
8.3.2	JDBC URL	(166)
8.3.3	连接到数据库	(166)
8.3.4	语句	(167)
8.3.5	ResultSet 对象	(168)
8.3.6	PreparedStatement 对象	(170)
8.3.7	CallableStatement 对象	(172)
第九章	ASP 与 ADO 数据库连接	(174)
9.1	ASP 概述	(174)
9.1.1	ASP 简介	(174)
9.1.2	ASP 的运行环境	(176)
9.1.3	ASP 文件及其编程特点	(176)
9.1.4	ASP 的内置对象和应用组件	(177)
9.1.5	ADO——ASP 的数据库存取组件	(177)
9.1.6	ASP 和传统 CGI 的对比	(178)
9.2	ASP 脚本语言介绍	(179)
9.2.1	VBScript 代码的基本格式	(179)
9.2.2	VBScript 变量	(180)
9.2.3	VBScript 过程和函数	(182)
9.2.4	数据类型	(184)
9.2.5	VBScript 的常数	(185)
9.2.6	VBScript 的基本语法	(185)
9.2.7	VBScript 的编码约定	(198)
9.2.8	ASP 基础	(202)
9.3	利用 ADO 来实现对数据库的访问	(219)
9.3.1	ADO 的原理	(219)
9.3.2	用 ADO 进行数据库编程概述	(220)
9.3.3	开始使用 ADO	(221)
9.3.4	Connection 对象	(224)
9.3.5	Command 对象	(233)
9.3.6	Parameters 集合	(238)
9.3.7	Parameter 对象	(239)

9.3.8	Errors 集合和 Error 对象	(240)
9.3.9	利用 GetString 函数来优化数据库编程	(241)
第十章	数据库管理系统简介	(243)
10.1	关系数据库管理系统产品概述	(243)
10.2	Oracle	(245)
10.2.1	Oracle 公司简介	(245)
10.2.2	Oracle 系统简介	(246)
10.2.3	Oracle 数据库系统的体系结构	(247)
10.2.4	数据库和实例的启动和关闭	(255)
10.2.5	数据字典的使用	(256)
10.2.6	事务管理	(257)
10.3	SQL Server	(258)
10.3.1	SQL Server 简介	(258)
10.3.2	SQL Server 的特点	(258)
10.3.3	SQL Server 结构	(262)
10.4	Sybase	(265)
第十一章	数据库新技术	(271)
11.1	数据库技术新发展	(271)
11.2	面向对象数据库系统	(274)
11.2.1	面向对象程序设计方法	(275)
11.2.2	面向对象数据模型	(276)
11.2.3	面向对象数据库语言	(277)
11.2.4	对象—关系数据库	(277)
11.3	并行数据库系统	(278)
11.3.1	并行数据库系统概述	(278)
11.3.2	并行数据库系统的目标	(279)
11.3.3	支持并行数据库的并行结构	(279)
11.4	多媒体数据库	(281)
第十二章	分布式数据库系统	(286)
12.1	概述	(286)
12.1.1	分布式数据库系统的定义	(286)
12.1.2	分布式数据库系统的特点	(286)
12.1.3	分布式数据库系统的目标	(287)
12.2	数据分布策略	(288)
12.2.1	数据分布的目的	(288)
12.2.2	数据分布的方式	(288)
12.2.3	关系的分割	(289)
12.2.4	数据分布带来的问题	(290)
12.3	分布式数据库系统的体系结构	(291)

12.3.1	分布式数据库系统的结构	(291)
12.3.2	分布式数据库管理系统	(291)
12.4	分布式数据库的发展前景和应用趋势	(293)
第十三章	数据仓库	(294)
13.1	概述	(294)
13.1.1	数据仓库的概念	(294)
13.1.2	数据仓库的好处	(294)
13.1.3	开发和管理数据仓库的问题	(295)
13.2	数据仓库框架	(295)
13.3	数据仓库的基本数据模式	(297)
13.4	数据仓库的基本操作	(298)
13.5	联机分析处理 (OLAP) 简介	(299)
13.5.1	OLAP 的体系结构	(300)
13.5.2	OLAP 的数据组织模式	(301)
13.5.3	实现的关键技术	(301)
13.5.4	OLAP 的局限性	(302)
13.5.5	总结	(303)
参考文献	(304)
后 记	(305)
互联网数据库自学考试大纲	(307)

第一章 绪 论

随着计算机技术、通信技术和网络技术的飞速发展，信息系统渗透到社会的各个领域，作为其核心和基础的数据库技术也得到了越来越广泛的应用。数据库的建设规模、数据库信息量的大小和使用频度已成为衡量一个国家信息化程度的重要标志。

20 世纪 90 年代以来，Internet 日益普及，Web 成为最流行、最大的网络信息系统，并以惊人的速度继续发展。Web 技术和数据库技术的结合，产生了互联网数据库这一新兴的数据库应用领域。

1.1 数据管理技术的发展

数据库技术是随着数据管理的需要而产生的。数据管理指的是对数据的分类、组织、编码、储存、检索和维护，它是数据处理的核心。随着计算机硬件和软件技术的发展，数据管理经历了如下三个阶段：①人工管理阶段；②文件系统阶段；③数据库管理阶段。

1.1.1 人工管理阶段

这一阶段即 20 世纪 50 年代中期以前，当时计算机主要用于科学计算，计算机的硬件状况是：外存只有磁带、卡片、纸带，没有磁盘等直接存取的存储设备；从软件看，没有操作系统，没有管理数据的软件，数据处理方式是批处理。

该阶段数据管理的特点是：

(1) 数据不保存。因为计算机主要应用于科学计算，一般不需要将数据长期保存，只是在计算某一具体实例时将数据输入，用完就撤走，不仅对用户数据如此处理，对系统软件有时也是这样。

(2) 数据需要由应用程序自己进行管理。应用程序不仅要规定数据的逻辑结构，而且还要设计物理结构，包括存储结构、存取方法、输入输出方式等。因此程序中存取数据的子程序随着存储的改变而改变，即数据与程序不具有独立性。这样不仅必须花费许多精力在数据的物理布置上，而且数据在存储上一有改变，就必须修改程序。

(3) 这一时期基本上没有文件概念，数据的组织方式必须由程序员自行设计。

(4) 数据不共享。一组数据对应一个程序，数据是面向应用的。即使两个应用程序涉及某些相同的数据，也必须各自定义，无法互相利用，互相参照，所以程序与程序之间有大量冗余数据。

上述特点如图 1-1 所示。

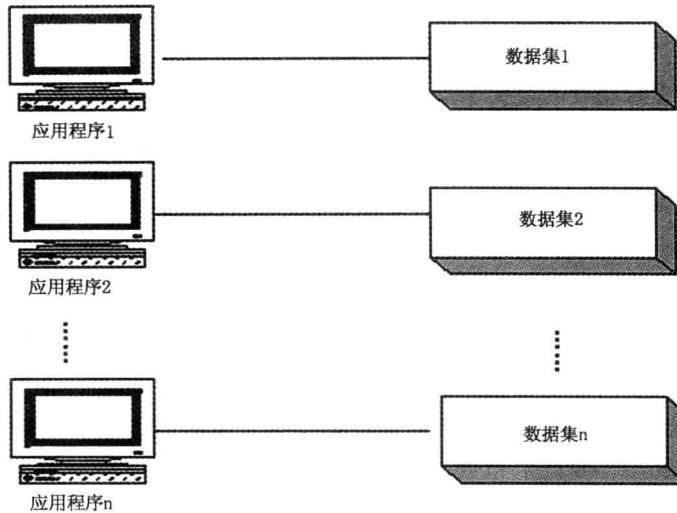


图 1-1

1.1.2 文件系统阶段

这一阶段从 20 世纪 50 年代后期到 60 年代中期，计算机硬件和软件都得到了发展。计算机不仅用于科学计算，还大量用于管理。这时硬件方面已经有了磁盘、磁鼓等直接存取的存储设备。在软件方面，操作系统中已经有了专门的数据管理软件，一般称为文件系统，处理方式上不仅有了文件批处理，而且能够联机实时处理。

该阶段的数据管理形成了如下几个特点：

(1) 数据可以长期保存。由于计算机大量用于数据处理，数据需要长期保留在外存上反复处置，即经常对文件进行查询、修改、插入和删除等操作。

(2) 文件系统管理数据。由于有软件进行数据管理，程序和数据之间有软件提供存取方法进行转换，有共同的数据查询修改的管理模块。文件的逻辑结构与存储结构由系统进行转换，使程序与数据有了一定的独立性。这样程序员可以集中精力于算法，而不必过多地考虑物理细节。并且，数据在存储上的改变不一定反映在程序上，这又可以大大节省维护程序的工作量。

(3) 文件已经多样化。由于已有了直接存取存储设备，也就有了索引文件、链接文件和直接存取文件等，而且用上倒排文件进行多码检索。

(4) 数据的存取基本上以记录为单位。

(5) 文件系统仍存在很多缺点，主要有：

① 数据共享性差，数据冗余度大。文件系统中文件基本上对应于某个应用程序，也就是说，数据还是面向应用的。当不同的应用程序所需要的数据有部分相同时，也必须建立各自的文件，而不能共享相同的数据。因此，数据冗余度大，浪费存储空间，并且由于相同数据的重复存储，各自管理，给数据的修改和维护带来了困难，容易造成数据的不一致性。

② 数据和程序缺乏独立性。文件系统中文件是为某一特定应用服务的。文件的逻辑结构

对该应用程序来说是优化的，因此要想对现有的数据再增加一些新的应用是很困难的，系统不容易扩充。一旦数据的逻辑结构改变，必须修改应用程序，修改文件结构的定义。而应用程序的改变，如应用程序所使用的高级语言的变化等，也将影响文件的数据结构的改变，数据和程序缺乏独立性。因此，文件系统仍然是一个不具有弹性的无结构的数据集合，所谓无结构，是指文件之间是孤立的，不能反映现实世界事物之间的内在关系。

这个时期程序与数据的关系如图 1-2 所示。

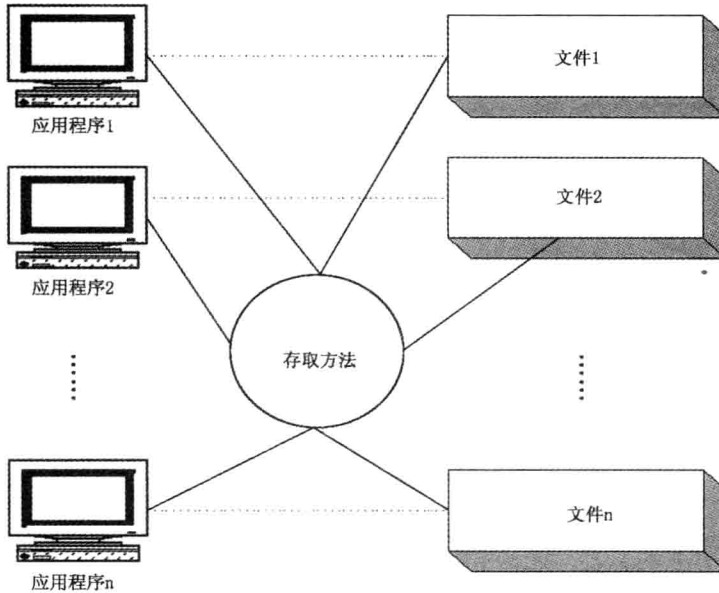


图 1-2

1.1.3 数据库系统阶段

20 世纪 60 年代后期以来，计算机硬件和软件技术得到了飞速发展；计算机用于管理的规模更为庞大，应用越来越广泛，数据量急剧增长，而且数据的共享要求越来越强。这种共享的含义是多种应用、多种语言互相覆盖地共享数据集合；有了大容量的磁盘；联机实时处理要求更多了，并开始提出和考虑分布处理；软件价格上升，硬件价格下降，为编制和维护系统软件及应用程序所需的成本相对增加。

为了解决多用户、多应用共享数据的需求，使数据为尽可能多的应用服务，相应出现了数据库这样的数据管理技术，它的出现使信息系统的研制从围绕加工数据的程序为中心转变到围绕共享的数据库来进行。这样既便于数据的集中管理，也有利于应用程序的研制和维护，提高了数据的利用率和相容性，从而提高了作出决策的可靠性。因此，大型复杂的信息系统大多以数据库为核心，数据库系统在计算机应用中起着越来越重要的作用。该阶段应用程序与数据之间的对应关系如图 1-3 所示。

数据库系统的特点如下：

(1) 数据结构化。

在文件系统中，相互独立文件记录内部结构的最简单形式是等长同格式记录的集合。例

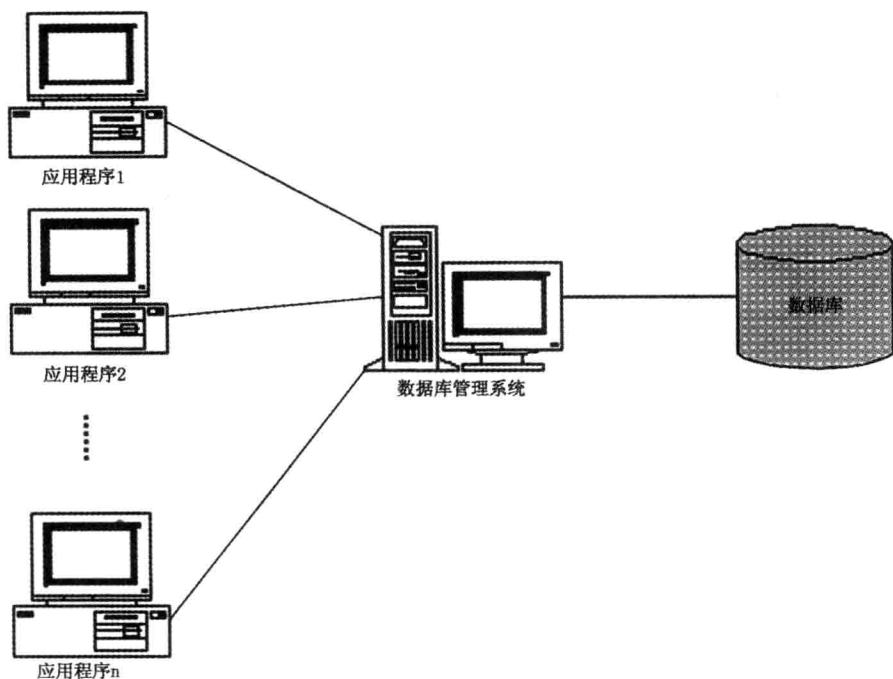


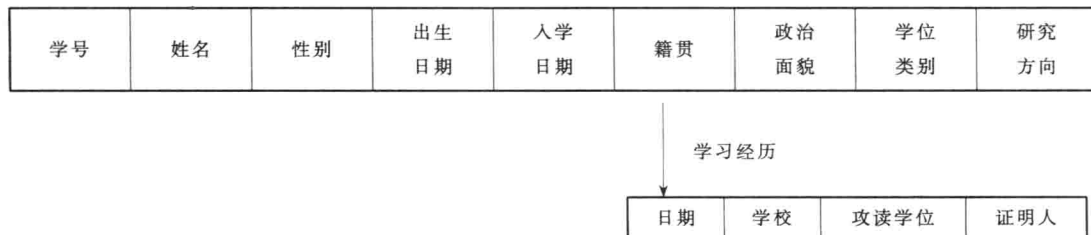
图 1-3

如一个研究生的基本概况文件，每条记录的记录格式如表 1-1。

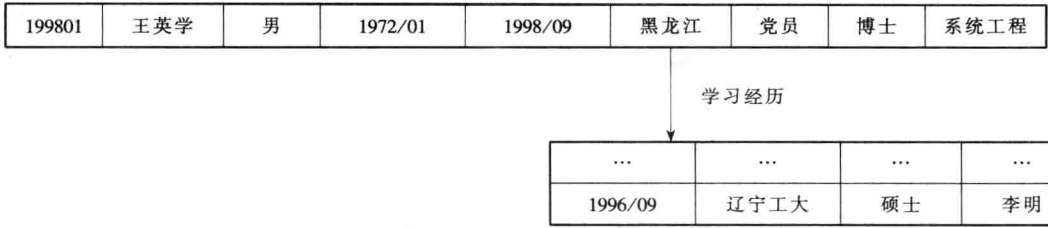
表 1-1 研究生基本概况记录

学号	姓名	性别	出生日期	入学日期	籍贯	政治面貌	学位类别	研究方向	学习经历
----	----	----	------	------	----	------	------	------	------

表中学习经历这栏内各个研究生所具有的信息量的多少变化很大。若以等长记录形式存储时，每个研究生记录的长度必须等于信息量最多的记录的长度，这样会造成存储空间的大量浪费。因此最好采用不等长记录（变长记录）或主记录与详细记录相结合的方法建立文件。在基本概况记录中，到研究方向为止的九个数据项是任何研究生必须具有而且是等长的，将这一部分作为主记录，而学习经历作为详细记录。这样，一个研究生基本概况记录的结构可如图 1-4 所示。图 1-4 (a) 表示每个记录的记录格式，而图 1-4 (b) 表示研究生基本概况记录的某一个记录值。这样由一个主记录附加几种各有若干个值的详细记录的方法增加了灵活性，节省了存储空间。



(a) 主记录——详细记录格式示例



(b) 学生记录示例

图 1-4 主记录——详细记录

上述思想是数据库方法的雏形，它已经把文件系统中记录内部有结构的思想扩大到了两个记录型之间。但这种方法还存在着局限性，因为这种灵活性只是对某一个应用而言的。而一个组织或部门包括许多应用，从整体观点来看，不仅要考虑一个应用（程序）的数据结构，而且要考虑整个组织的数据结构问题。整个组织的数据结构化，这就要求在描述数据时不仅描述数据本身，还要描述数据之间的联系。文件系统中尽管记录内部已有了某些结构，但记录之间是没有联系的，孤立的。因此，数据的结构化是数据库主要特征之一，是数据库与文件系统的根本区别。

(2) 数据共享性高、冗余度小、易扩充。

数据库从整体观点来看待和描述数据，数据不再是面向某一应用（程序），而是面向整个系统的，这可以大大减小数据的冗余度，即节约存储空间，减少存取时间，又可避免数据之间的不相容性和不一致性。

对数据库数据的应用可以有很灵活的方式，可以取整体数据的各种合理子集用于不同的应用系统，而且当应用需求改变或增加时，只要重新选取不同子集或者加上一小部分数据，便可以有更多的用途，满足新的要求。这就是弹性大，易扩充的特点。

(3) 数据独立性高。

数据独立性是数据库领域的一个常用术语，包括数据的物理独立性和数据的逻辑独立性。数据库系统提供了两方面的映象功能。一个是数据的存储结构与逻辑结构之间的映象或转换功能，一个是数据的总体逻辑结构与某类应用所涉及的局部逻辑结构之间的映象或转换功能。

第一种映象功能使得当数据的存储结构（或物理结构）改变时，数据的逻辑结构可以不变，从而应用程序也不必改变。这就是数据和程序的物理独立性。简称数据的物理独立性。

数据库系统中某一类应用使用的数据通常是总体数据的子集，而且各类应用对同一数据的使用要求也不一定相同。数据库系统通常提供局部数据结构的说明功能。局部的数据结构可以按具体应用要求作一定的改变。系统提供对这些改变的映象和转换功能（即上面的第二种映象功能）。使得当总体逻辑结构改变时，通过对映象的相应改变而保持局部逻辑结构不变。程序员是根据局部逻辑结构编写应用程序的，因而应用程序也就可以不必改变。这就是数据和程序的逻辑独立性。简称数据的逻辑独立性。

数据和程序的独立性，把数据的定义和描述从应用程序中分离出去。此外，数据的存取又有 DBMS 管理，用户不必考虑存取路径等细节，从而简化了应用程序的编制，大大减少了应用程序的维护和修改的次数。

(4) 统一的数据管理和控制。