

数据库原理与 SQL Server 2000 教程

苏 啸 编著

北京工业大学出版社

内 容 提 要

全书共十五章：第一章和第二章介绍数据库系统基本概念、数据模型，第三章和第四章较详细地讲述了关系数据理论基础，数据库设计。后十一章为 SQL Server 2000。

本书采取了理论和实践相结合的方式，一方面详细阐述了数据库的基本理论和方法、详细介绍了 SQL Server 2000 的各种功能和特点。另一方面注重培养读者解决实际问题的能力，通过对专题内容的详细理论讲解和实际示例说明，使读者在学习完本书之后可以更加深刻地理解数据库的基本理论方法和 SQL Server 2000 的各种功能用法。

本书适用于本科、专科高等院校和高职的计算机专业、信息管理专业、电子商务专业的学生作为教材和参考书，也可供从事计算机与信息工作的科技人员学习参考。

图书在版编目 (CIP) 数据

数据库原理与 SQL Server 2000 教程/苏啸编著. —北京：北京工业大学出版社，2002.10
ISBN 7-5639-1186-3

I. 数… II. 苏… III. 关系数据库 - 数据库管理系统, SQL Server 2000 - 高等学校 - 教材 IV. TP311.138

中国版本图书馆 CIP 数据核字 (2002) 第 077601 号

数据库原理与 SQL Server 2000 教程

苏 啸 编著

※

北京工业大学出版社出版发行

邮编：100022 电话：(010) 67392308

各地新华书店总经销

徐水宏远印刷厂印刷

※

2002 年 10 月第 1 版 2002 年 10 月第 1 次印刷
787 × 1092 毫米 16 开本 23.25 印张 580 千字

印数：0001 ~ 3000 册

ISBN 7-5639-1186-3/T·193

定价：37.00 元

前 言

本书是作者在多年从事数据库教学和科研的基础上编写出来的,是一本面向本科、专科高等院校和高职的计算机专业、信息管理专业、电子商务专业的学生而编写的教材和参考书,也可供从事计算机与信息工作科技人员参考。该书是一本理论与实用相结合、较前沿的专业用书及参考资料。

数据库技术是计算机与信息科学中发展最快、应用最广泛的技术之一。从 20 世纪 60 年代末开始在 30 年的发展历程中,它已经历了第一代层次、网状数据库系统,第二代关系数据库系统,到正在研究的新一代数据库系统。数据库技术已成为当今计算机信息系统的基础和核心,要学习计算机科学,就不能不学习掌握数据库技术。

作为一本教程,本书采取了理论和实践相结合的方式,一方面详细阐述了数据库的基本理论和方法、详细介绍了 SQL Server 2000 的各种功能和特点。另一方面注重培养读者解决实际问题的能力,通过对专题内容的详细讨论讲解和实际示例说明,使读者在学习完本书之后可以更加深刻地理解数据库的基本理论、方法和 SQL Server 2000 的各种功能用法。

SQL Server 2000 是新一代的数据库和数据分析系统,可以帮助用户建立 Internet 商务体系,是一套完全的数据库和数据分析解决管理软件,该产品可以使用户快速地创建新一代的可扩展电子商务和数据仓库解决方案。SQL Server 2000 具有完全的 Web 功能,支持扩展标记语言 (XML)。SQL Server 2000 采用了业界先进的数据结构,能够为要求苛刻的 Web 应用和企业应用提供高度的可扩展性和高可靠性。在该产品中,微软公司提供了许多技术和工具,作为数据库系统的标准集成组件,使得 SQL Server 2000 可以进行最快速的应用程序开发。Windows 2000 和 SQL Server 2000 群集提供了高度可用性和自动化管理。SQL Server 2000 还提供了强大的管理模型,它以图形化用户界面和向导工具为基础,带有一个用于重复任务执行的作业调度程序以及用于编写特定应用操作脚本的 SQL - DMO。

全书共十五章。前四章为数据库理论、方法,第一章和第二章介绍数据库系统基本概念、数据模型,第三章和第四章较详细地讲述了关系数据库理论基础、数据库设计。后十一章为 SQL Server 2000。第五章、第六章和第七章介绍了 SQL Server 的系统组成,SQL Server 的数据类型和 SQL Server 编程结构,第八章介绍 SQL Server 的安全性和用户管理,第九章和第十章介绍 Transact 数据定义语言 DDL, Transact 数据操纵语言 DML,第十一章和第十二章介绍设计数据完整性、存储过程与触发器,第十三章 SQL Server 2000 提供的应用程序接口,第

十四章 SQL Server 与 ADO，第十五章 SQL Server 2000 对 XML 的支持。

本书由苏啸编著，胡艳维、周锦春提出了许多宝贵的意见和建议，在此向他们表示衷心的感谢。我还要感谢我的家人，没有他们的支持、理解和帮助，这本书是不可能完成的。同时感谢在成书过程中所有支持、关心和帮助过我的人们，向他们表示深深的谢意。

由于作者水平有限，书中难免存在许多错误和不足之处，恳请读者批评指正。

苏 啸

2002 年 10 月

目 录

第1章 数据库系统基本概念	1	2.4 数据库类型	23
1.1 数据与信息处理与数据管理	1	2.4.1 层次模型(略)	23
1.1.1 数据与信息	1	2.4.2 网状模型(略)	23
1.1.2 数据处理与数据管理	2	2.4.3 关系模型	23
1.2 数据管理技术的发展	2	2.5 E-R方法	25
1.2.1 自由管理方式	2	2.5.1 问题的提出	26
1.2.2 文件管理方式	2	2.5.2 E-R图的基本图素及画法	26
1.2.3 数据库方式	4	2.5.3 怎样设计E-R图	27
1.3 数据库系统的组成与特点	5	2.6 从E-R图导出关系数据模型	29
1.3.1 系统的组成	5	2.7 小结	32
1.3.2 数据库系统的特点	6	第3章 关系数据库理论基础	34
1.4 数据库体系结构	8	3.1 关系的数学定义	34
1.4.1 数据库三级组织结构	8	3.1.1 引例	34
1.4.2 三个模式之间的映射	9	3.1.2 定义	35
1.4.3 用户与数据库接口	10	3.2 关系的性质	36
1.5 DBMS组成	10	3.3 关系数据库描述	37
1.5.1 语言	11	3.3.1 问答式	37
1.5.2 数据库运行程序	11	3.3.2 语言描述式	37
1.5.3 数据库维护程序	11	3.4 关系数据库操作语言DML	39
1.6 数据库保护	12	3.4.1 关系DML特点	39
1.6.1 安全性保护	12	3.4.2 关系代数	40
1.6.2 完整性保护	12	3.4.3 关系演算	44
1.6.3 故障恢复	13	3.4.4 关系数据操作语言的完备性	45
1.6.4 并发控制	13	3.5 关系数据库标准语言——SQL	45
1.6.5 死锁问题	14	3.5.1 SQL语言的特点	45
1.7 小结	15	3.5.2 SQL的数据定义语言(DDL)	46
第2章 数据模型	17	3.5.3 SQL的数据操纵语言(DML)	47
2.1 什么是数据模型	17	3.6 关系规范化	52
2.2 实体和属性	18	3.6.1 准备知识	52
2.2.1 实体	18	3.6.2 范式与规范化	54
2.2.2 属性	18	3.6.3 分解关系的基本原则	60
2.2.3 实体和属性的型与值	19	3.7 小结	61
2.2.4 关键字	19	第4章 数据库设计	65
2.2.5 信息转化为数据	19	4.1 数据库设计的内容及特点	65
2.3 实体间的联系	20	4.1.1 数据库设计内容	65
2.3.1 数据的描述	20	4.1.2 数据库设计的特点	65
2.3.2 数据联系方式	21	4.2 数据库逻辑设计	66
2.3.3 实体联系的表示法	22	4.2.1 逻辑设计步骤	66
2.3.4 实际可能存在的实体联系情况	22		

4.2.2 第一阶段设计任务——收集和分析 用户要求	67	6.1.3 字符串与二进制类型	113
4.2.3 第二阶段设计任务——建立 E-R 模型	71	6.1.4 专有类型	113
4.2.4 第三阶段设计任务——数据库模式 设计	72	6.2 SQL Server 构架	114
4.3 数据库物理设计	72	6.2.1 数据类型和表结构	114
4.4 应用程序与系统的运行和维护	73	6.2.2 空值	115
4.4.1 应用程序设计	73	6.2.3 系统表	115
4.4.2 投入运行和维护	74	6.2.4 临时表	115
4.5 编写技术文档	74	6.2.5 使用表	116
4.5.1 系统说明书	74	6.2.6 用户定义的数据类型	117
4.5.2 技术说明书	75	6.3 小结	118
4.5.3 使用说明书	75	第 7 章 SQL Server 编程结构	120
4.6 信息系统开发	75	7.1 程序注释语句	120
4.6.1 调查研究阶段	75	7.2 批处理	121
4.6.2 系统分析阶段	77	7.3 局部变量	124
4.6.3 系统设计阶段	79	7.3.1 声明局部变量	124
4.6.4 系统实施阶段	81	7.3.2 为局部变量赋值	124
4.6.5 系统维护评价阶段	82	7.4 全局变量	127
4.7 小结	83	7.5 条件判断结构语句	131
第 5 章 SQL Server 2000 的系统组成	85	7.5.1 IF...ELSE	131
5.1 SQL Server 2000 的系统数据库	85	7.5.2 BEGIN...END 语句块	132
5.2 SQL Server 2000 的系统表	86	7.5.3 WHILE 循环结构	133
5.3 SQL Server 2000 的存储过程	88	7.5.4 CASE	134
5.4 SQL Server 2000 的示例数据库	101	7.5.5 WaitFor 语句	135
5.5 SQL Server 2000 的函数	103	7.5.6 RETURN	135
5.5.1 合计函数	104	7.6 游标	135
5.5.2 数据转换函数	104	7.6.1 游标实现	136
5.5.3 游标函数	104	7.6.2 声明游标	136
5.5.4 日期和时间函数	104	7.6.3 打开游标	140
5.5.5 数学函数	105	7.6.4 关闭游标	141
5.5.6 元数据函数	106	7.6.5 释放游标	142
5.5.7 安全性函数	107	7.6.6 使用游标取数	142
5.5.8 字符串函数	107	7.6.7 利用游标修改数据	146
5.5.9 系统函数	108	7.6.8 游标参数和变量	148
5.5.10 文本和图像函数	109	7.7 事务	149
5.5.11 配置函数	109	7.7.1 事务的概念	149
5.6 小结	110	7.7.2 事务的工作原理	149
第 6 章 SQL Server 数据类型	111	7.7.3 使用事务的考虑	150
6.1 数据类型	111	7.8 小结	152
6.1.1 精确数字类型	111	第 8 章 安全性和用户管理	153
6.1.2 近似数字类型	112	8.1 SQL Server 2000 安全体系结构	153
		8.1.1 概述	153
		8.1.2 安全账户认证	154
		8.2 设计安全策略	156

8.3 SQL Server 2000 用户和固定角色	157	9.6 创建视图举例	188
8.4 用户管理	159	9.6.1 复合条件	188
8.4.1 存储过程管理用户	159	9.6.2 重新排序和重新命名列	189
8.4.2 SQL Server 企业管理器管理用户	160	9.6.3 导出列	189
8.5 角色管理	162	9.6.4 自连接表	189
8.6 权限管理	163	9.6.5 连接多个表	190
8.6.1 SQL Server 的权限	163	9.7 创建索引	191
8.6.2 使用 Transact-SQL 语句管理权限	165	9.7.1 聚族索引	192
8.7 SQL Server 的审计	166	9.7.2 其他索引选项	194
8.8 小结	167	9.8 删除数据库、表、视图和索引	195
第 9 章 Transact 的数据定义语言		9.9 小结	195
DDL	168	第 10 章 Transact 数据操纵语言	
9.1 创建数据库	168	DML	197
9.1.1 指定位置和大小	168	10.1 Select 语句	197
9.1.2 修改数据库	169	10.1.1 搜索条件	199
9.1.3 定义文件组	170	10.1.2 三值逻辑	199
9.2 创建表	171	10.1.3 从视图中检索数据	200
9.2.1 创建表的基本要求	171	10.1.4 指定要检索的列	201
9.2.2 指定文件组上创建表	173	10.1.5 剔除重复的行	201
9.3 表的特殊性列	173	10.1.6 常量、函数和表达式	201
9.3.1 Time Stamp 列	173	10.2 Select 的条件	201
9.3.2 Identity 列	174	10.2.1 Basic 条件	202
9.3.3 行全局惟一标识符列	174	10.2.2 Null 条件	202
9.3.4 Sql_variant	174	10.2.3 Between 条件	202
9.3.5 Table 型数据与用户自定义函数	175	10.2.4 In 条件	203
9.3.6 空列和非空列	176	10.2.5 Like 条件	203
9.3.7 用户定义的数据类型	177	10.2.6 Exists 条件	205
9.3.8 缺省值	177	10.2.7 限定的条件	207
9.3.9 计算的列	178	10.3 Select 的子句	208
9.3.10 增加、删除和修改表列	178	10.3.1 Group By 子句	208
9.4 表的约束	179	10.3.2 Having 子句	210
9.4.1 非空约束	179	10.3.3 Order By 子句	211
9.4.2 键约束	179	10.4 复杂的 Select 语句	212
9.4.3 检查约束	181	10.4.1 使用联接 (在 From 子句中指定多 个表)	212
9.4.4 惟一性约束	181	10.4.2 Union 关键字	217
9.4.5 禁止用于复制的约束	182	10.5 Select 的其他用法	219
9.4.6 增加、删除和禁止约束	182	10.5.1 使用 Case 表达式	219
9.5 创建视图	183	10.5.2 使用子查询作为标量值	220
9.5.1 定义视图内容	184	10.5.3 在 From 子句中使用 select 表 达式	220
9.5.2 选择表达式	184	10.5.4 其他 Select 语句功能	221
9.5.3 合并多个表	184		
9.5.4 视图分类	186		
9.5.5 创建视图选项	187		

10.5.5 Group 子句的 Rollup 和 Cube 选项	222	12.2.3 执行存储过程	262
10.5.6 Compute 子句	223	12.2.4 删除存储过程	269
10.5.7 Into 子句	223	12.2.5 修改存储过程	269
10.6 Insert 语句	224	12.2.6 通过存储过程传递参数	269
10.6.1 基本 Insert 语句	224	12.2.7 确定存储过程的执行状态	271
10.6.2 多行 Insert 语句	224	12.3 存储过程错误处理	271
10.6.3 在视图中插入行	225	12.3.1 调用 RaisError 语句	271
10.7 Update 语句	226	12.3.2 调用 sp_addressage 系统存储 过程	272
10.7.1 在 Update 语句中使用子查询	226	12.4 触发器概述	273
10.7.2 修改多个表	227	12.5 触发器使用与管理	274
10.8 Delete 和 Truncate Table 语句	228	12.5.1 创建触发器	274
10.8.1 清除整个表	229	12.5.2 前触发器	277
10.8.2 从多个表中删除行	229	12.5.3 管理触发器	277
10.9 并行修改和表锁	229	12.5.4 删除触发器	278
10.9.1 如何防止访问冲突	229	12.5.5 修改触发器	278
10.9.2 如何维持数据库的一致性	230	12.6 触发器的特殊功能	279
10.10 小结	232	12.6.1 inserted 和 deleted 表	279
第 11 章 设计数据完整性	234	12.6.2 建立列级触发器	280
11.1 数据完整性基础	234	12.6.3 检查指定列的变化	281
11.1.1 数据完整性简述	234	12.7 利用存储过程和触发器维护数据 完整性	282
11.1.2 数据完整性分类	235	12.8 小结	283
11.1.3 实施数据完整性的途径	235	第 13 章 SQL Server 2000 提供的应用程序 接口	285
11.2 使用约束实施数据完整性	236	13.1 SQL Server 应用程序接口概述	285
11.2.1 使用 PRIMARY KEY 约束	237	13.1.1 一般数据库访问 API	285
11.2.2 使用 UNIQUE 约束	239	13.1.2 SQL Server 数据库服务 API	286
11.2.3 使用 CHECK 约束	241	13.2 ADO 与 SQL Server	286
11.2.4 使用 FOREIGN KEY 约束	243	13.2.1 OLE DB 提供者	287
11.2.5 使用数据库关系图管理外键 约束	246	13.2.2 在不同的开发环境下使用 ADO	287
11.3 使用规则	248	13.3 OLE DB 与 SQL Server	288
11.3.1 使用 Transact - SQL 管理规则	248	13.3.1 SQL Server OLE DB 提供者的 安装	288
11.3.2 使用企业管理器管理规则	250	13.3.2 开发 OLE DB 应用程序	289
11.4 使用默认值	251	13.4 ODBC 与 SQL Server	290
11.4.1 使用 Transact - SQL 管理默认值	251	13.5 用于 Visual Basic 的 DB - Library	291
11.4.2 使用企业管理器管理默认值	252	13.6 SQL - DMO	292
11.5 使用 IDENTITY 列	252	13.6.1 SQL - DMO 语约定	292
11.6 小结	254	13.6.2 SQL - DMO 对系统的要求	293
第 12 章 存储过程与触发器	256	13.6.3 安装 SQL - DMO	293
12.1 存储过程概述	256	13.7 复制编程	294
12.2 存储过程的使用和管理	257	13.7.1 复制编程的优点	295
12.2.1 创建存储过程	257		
12.2.2 管理存储过程	260		

13.7.2 复制编程的计划	295	15.1.1 IIS 虚拟目录管理的系统要求	330
13.8 DTS 编程	296	15.1.2 使用 SQL Server 的 IIS 虚拟目录管理工具	330
13.9 扩展存储过程编程	297	15.1.3 创建虚拟目录	331
13.10 URL 访问与 SQL Server	297	15.2 SQL 对象模型的 IIS 虚拟目录管理	333
13.11 小结	299	15.2.1 SQLVDirControl 对象	334
第 14 章 SQL Server 与 ADO	300	15.2.2 SQLVDirs 集合对象	335
14.1 ADO 对象模型	300	15.2.3 SQLVDir 对象	335
14.1.1 Connection 对象	301	15.2.4 VirtualNames 集合对象	337
14.1.2 Recordset 对象 (记录集)	303	15.2.5 VirtualName 对象	337
14.1.3 Errors 集合和 Errors 对象	304	15.2.6 使用对象模型创建 nwind 虚拟目录	338
14.1.4 Parameters 集合和 Parameter 对象	305	15.3 使用 HTTP 访问 SQL Server 服务器	339
14.1.5 Fields 集合和 Field 对象	307	15.3.1 基本概念	339
14.1.6 Stream 对象	308	15.3.2 XML 文档与文档碎片	340
14.1.7 Properties 集合和 Property 对象	309	15.3.3 三层系统体系结构	340
14.2 不同开发环境中的 ADO 应用	310	15.3.4 特定的字符	341
14.2.1 Visual Basic 与 ADO	310	15.3.5 URL 编码中的实体编码	342
14.2.2 Visual C++ 与 ADO	310	15.4 使用 HTTP 执行 SQL 语句	342
14.2.3 Web-Based 应用	311	15.5 使用 HTTP 执行存储过程	346
14.3 使用 ADO 执行查询	311	15.6 使用 HTTP 访问模板文件	348
14.3.1 数据源管理	311	15.6.1 使用 XML 模板	348
14.3.2 使用 ADO 连接 SQL Server 数据源	311	15.6.2 使用模板执行 SQL 查询	350
14.3.3 执行查询	313	15.6.3 向模板传递参数	352
14.4 处理结果	319	15.6.4 在模板中指定 XSL 样式表	355
14.5 ADO 高级特性	320	15.7 SQL Server 的 OLE DB 提供者对 XML 的扩展	357
14.5.1 在 ADO 中使用游标	320	15.7.1 使用 IcommandText 设置 XML 命令	357
14.5.2 在 ADO 中执行事务	321	15.7.2 使用 IcoammandStream 设置 XML	357
14.5.3 在 ADO 中处理 DDL	322	15.7.3 SQL OLE DB 提供者的特定属性	358
14.5.4 管理长数据类型	323	15.8 POST 模板的示例应用程序	359
14.5.5 使用 ADO Stream 对象访问并修改 SQL Server BLOB 数据	325	15.8.1 使用 HTML 表单发送模板	359
14.6 小结	327	15.8.2 直接向虚拟目录发送模板	361
第 15 章 SQL Server 2000 对 XML 的支持	329	15.9 小结	361
15.1 使用 IIS 虚拟目录管理 SQL Server 应用	329		

第1章 数据库系统基本概念

数据库是一门研究数据管理的技术，始于 60 年代末，经过 30 多年的发展，已自成理论体系，成为计算机软件的一个重要分支。数据库技术体现了当代先进的数据管理方法，赢得社会的广泛承认，使计算机应用真正渗透到国民经济各部门，在信息社会中正发挥越来越大的作用。

本章从数据管理技术的发展，引出数据库概念，围绕着数据库系统，介绍有关名词术语，从而为读者建立起数据库系统的基本概念。

1.1 数据与信息处理与数据管理

数据是数据库系统研究和处理的对象。信息是以人们需要组合并使用的数据。信息一般是靠数据来描述，所以数据与信息是分不开的，它们既有联系又有区别，因此首先得搞清数据与信息概念上的不同。

计算机的出现，开辟了高速数据处理的新纪元。电子数据处理中的核心问题是数据管理，即怎样有效地管理大量的数据，如何合理地组织、存储、检索、维护数据，是使用数据库系统必须完美解决的问题。

1.1.1 数据与信息

数据，本质上是对客观事物特征的一种抽象的、符号化的表示。即用一定的符号表示那些从观察或测量中所收集到的基本事实，采用什么符号完全是一种人为的规定。如在计算机内部用 0 和 1 两个符号描述事物的特征。值得注意的是，不能把数据仅仅理解成“数量”概念，它有着更广泛的含义，除了量化特征描述外，通常还需要一些非量化特征的描述。例如，气象预报中，温度的高低固然可以量化地表示，而“刮风”或“下雨”等特征则需用文字或图形符号进行描述，后者也是数据，但数据类型不同。

未经处理的数据只是基本素材，当对其进行适当地加工处理，产生出有助于实现特定目标的信息后对人们才有意义。可见，信息实际上是指经过加工处理后的数据，是整理消化后的数据。例如，“除去物价上涨因素，本市今年生活指数较去年同期提高了 8 个百分点”。这是一条信息，它的产生是经统计分析大量原始数据资料后得出的结论，而其在理论意义上虽然也是数据，但含义不同了。

综合所述，数据与信息两者密不可分，既有联系，又有区别。数据表示了信息，而信息只有通过数据形式表示出来才能被人理解和接受。尽管数据与信息两者在概念上不尽相同，但通常人们并不严格去区分它们，通常所说的数据处理也可称为信息处理。

信息资源，可以被利用，并影响人们的行为动作。人类进行各项社会活动，不仅要考虑物质条件，而且要认真研究信息，实际上人们的活动总是伴随着数据处理的全过程，人们越来越认识到信息的重要性，在知识爆炸的今天，人们将物质、能量与信息并列为人类社会活动的三个基本条件。人类社会正在由工业社会时代进入信息社会，知识经济时代。

1.1.2 数据处理与数据管理

数据处理也可称为信息处理。因为当把客观事物描述表示成数据后，这些数据便被人们赋予了特定的含义，从而为人们提供了不必直接观察和度量事物就可获得有关信息的手段。

数据处理是从某些已知的数据出发，推导出一些新的数据，这些新的数据又表示了新的信息的过程。具体操作中，数据处理涉及到数据的取样、收集、管理、传送、加工利用乃至信息输出的演变与推导全过程。

通常，数据处理的计算方法和过程比较简单，但因其表现为数据量大，数据结构复杂，数据之间有复杂的逻辑联系。所以，数据处理业务的焦点不是计算，而是数据管理，指数据的收集，整理，组织、存储、维护、检索、传送等操作，这些操作是数据处理的基本环节，是任何数据处理业务中都必须有的共性部分；而加工和处理，则不同业务各不相同。因此，对数据管理部分，要加以突出，集中精力研制出一个通用，高效而又使用方便的管理软件，把数据有效地管理起来，以便最大限度的减轻程序员及用户的负担，至于处理业务中的加工计算，因不同业务各不相同，要靠程序员根据业务情况编写应用程序加以解决。

数据处理与数据管理密切相联，数据管理技术的优劣，将直接影响数据处理的效果，数据库技术正是针对这一目标研究、发展并逐渐完善起来的专门技术。

1.2 数据管理技术的发展

从 50 年代到现在，数据管理技术是一个快速发展的历程，大体上经历了三个发展阶段：自由管理方式、文件管理方式、数据库方式。

1.2.1 自由管理方式

最早的计算机，没有必要的软件支持，因此用户使用计算机进行数据处理不得不自行管理数据，程序员在程序中，既要考虑处理方法，又要管理数据的组织存储，如图 1-1 所示。

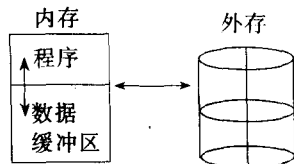


图 1-1 自由管理方式

存取数据是根据设备的物理地址进行的，这种方式迫使程序员直接与物理设备打交道。管理数据存放地址，导致程序与数据紧密相关，程序高度依赖于数据，数据稍有变动，整个程序就必须全部进行修改，编程效率低，程序不灵活而且容易出错。

1.2.2 文件管理方式

文件管理方式，是把有待于加工处理的数据组织成数据文件，文件可以命名，一旦命名之后，程序中便可以通过文件名逻辑性地存取文件中的数据，解脱了程序员直接与物理设备打交道的沉重负担。这二者间需要一个转换过程，但这是由文件管理系统完成的，程序中不

再涉及任何物理细节。由操作系统的知识我们知道，文件是操作系统管理数据的基本单位，文件管理系统负担数据的逻辑组织与物理组织之间的映射。

数据组织成文件后，使程序仅涉及文件中的数据，而与数据的物理存储位置无关，这是程序员所追求的，目的是克服数据任何存储位置的变动导致程序全部作废的弊病。

由于文件可以根据数据所代表的含义分门别类，按一定规则加以组织，所以文件中的数据能描述现实世界事物的特征。描述事物各种特征或性质的最小数据单位称数据项；若干数据项的集合称记录。记录能表达事物，刻画某种信息；相同性质的数据项构成的记录集合就称文件。

数据有规律地组织成文件形式存放，含义清楚，逻辑关系非常明确，使数据处理真正体现了信息处理，按名存取数据，既形象又方便。

文件是操作系统管理的基本单位。文件在存储介质上（如磁盘，光盘）的摆放以及如何读写是操作系统的事，不需要用户参与。但文件怎么组织（建立自己的目录树），则是用户的事。在文件方式下，每个用户都可以建立、维护或者处理一个或多个存放数据的文件。

文件管理方式，是数据管理的一大进步，直到今天，仍被广泛采用。即使是数据库方式，也以文件方式为基础。鉴于文件管理方式的重要性，下面举例说明：

设图 1-2 是假想的汽车运输公司的组织机构图，方框代表部门，方框间的连线表示部门之间有业务联系。

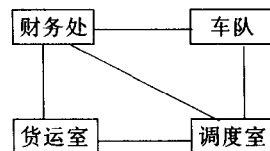


图 1-2 汽车运输公司体制

假定：

- ① 货运室处理货运业务，建立如表 1-1 所示的文件。

表 1-1 货运文件

单位	合同号	品名	数量	出发地	目的地
木工厂	990101	圆木	30	长平	上海
机械公司	990102	管材	80	新钢	南海

- ② 调度室处理调度业务，建立如表 1-2 所示的文件。

表 1-2 调度文件

车队号	品名	数量	出发地	目的地	货到时间
木工厂	圆木	16	长平	上海	2000-2-6
机械公司	管材	21	新钢	南海	2000-3-2

- ③ 财务科处理工资业务，建立如表 1-3 所示的文件。

表 1-3 工资文件

姓名	工资	奖金	补贴	总额
李国	700	130	400	1 230
王林	860	180	300	1 440

④ 车队处理派车业务，建立如表 1-4 所示的文件。

表 1-4 出车文件

出车日期	姓名	派车单号	车型	出发地	目的地	品名	数量
2000-1-28	李国	A9001	平板	长平	上海	圆木	16
2000-2-26	王林	A9002	东风	新钢	南海	管材	7
2000-2-26	陈江	A9003	东风	新钢	南海	管材	7
2000-2-26	钟卉	A9004	东风	新钢	南海	管材	7

这样，每个部门都建立有自己的文件，并各自处理和维持自己文件，在文件管理系统支持下按文件名存取数据进行加工计算，如图 1-3 所示。

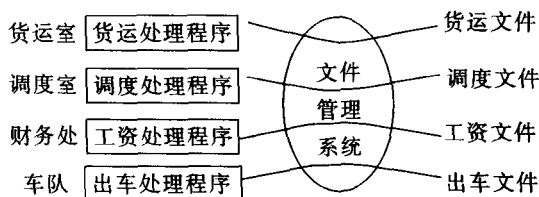


图 1-3 文件管理方式

文件管理方式为直接面向应用，有一个应用，必须有一个对应的文件（也可能多个）与之对应，文件由用户建立和维护。

文件管理方式并不合理，因为各个部门（用户）自行其是，互不通气自建数据文件。然而企业是个整体，各部门互有联系，数据要相互引用，但是由于文件面向应用，只能建立一一对应的关系，导致各部门不得不重复收集并存储大量相同数据，极大影响了文件管理方式的运用。

文件管理方式的不足之处是：

- ① 程序与文件相互依存。这说明程序仍与数据相关，文件有所变动，程序就要相应修改，而文件离开了使用它的程序便全部失去存在的价值。
- ② 数据冗余大。
- ③ 数据容易发生矛盾，因为同一个数据出现在不同文件中，随着数据更新，会造成同一个数据在不同文件中具有不同数值，这是数据冗余的恶果。
- ④ 不能反映企业间的业务联系。

由于企业各部门有着千丝万缕的联系，这种联系至关重要，数据要互相引用，但是文件方式面向应用组织文件，把至关重要的联系信息丢掉了。在此基础上产生了数据库方式。

1.2.3 数据库方式

针对文件方式的弊端，人们改进后推出了数据库方式。数据库方式的目标，首先是克服数据文件与处理程序间的相互依存，力求数据独立。其次建立数据之间的各种联系，大幅度消除数据冗余。再是着力解决数据的安全性、完整性等数据保护问题。

数据库方式与文件方式根本区别在于，前者面向系统，后者面向应用，即数据库方式综

合全体应用,用整体观点组织数据,形成一个有完整数据结构的集约化数据集合,这个数据集合实际上就是数据库。数据库方式的最大特点就是数据文件相对独立。

图1-4所表示的综合,并非数据的简单相加合并,而是通过规定的数据结构形式把各个文件中数据联系起来形成的统一整体,这个整体如同一个数据仓库,能从其中导出不同用户所需要的数据。

在数据库方式下,特定用户处理自己业务时,不需要自建文件,其所使用的数据是从事先经过严密设计的数据库中导出所需要的部分,并作为应用程序使用的“文件”,“文件”并非独立存在,更非实际存储,而是取自库中的某个子集,又因它是从数据库中映射出来的,所以这个文件叫逻辑文件,如图1-5所示。逻辑文件是在使用时生成,用完之后自动消失。

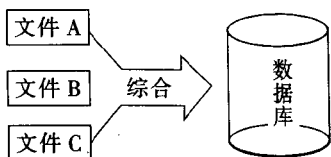


图 1-4 数据库构成

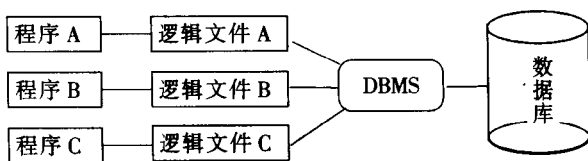


图 1-5 数据库方式工作示意图

图1-5说明应用程序要通过DBMS与数据库建立连接。DBMS是“数据库管理系统”的简称,全文是Database Management System。它是管理、控制数据库的软件,由DBMS负责数据库与逻辑文件的映射,用户只需指出映射要求。此外,在数据库方式下,逻辑文件的维护是由DBMS完成的,用户与程序员只需拿来就能用,不必像文件方式那样程序员自己维护文件,这减轻了程序员管理数据的负担,大大提高了编程效率。

1.3 数据库系统的组成与特点

数据库实际上是个系统,它由数据库和支持数据库运行的软、硬件环境以及应用程序和数据库管理员等相互独立而又相互联系的若干部分所组成。

1.3.1 系统的组成

1. 数据库

数据库是一个结构化的相关数据的集合。主要是通过综合汇总各个数据库用户的文件,除去不必要的冗余,然后使各个文件的数据项相互联系,从而形成整体数据结构。联系是数据库的重要特点,至于怎样实现这种联系,取决于数据库类型,类型不同,联系手段也不一样,这个问题在第二章详细讨论。数据库是集成化的,能为多种应用服务,它独立于应用程序,是数据处理系统的核心和管理对象。

2. 数据库管理系统 DBMS

在数据库系统中,DBMS是专门用于数据管理的软件。通常,DBMS提供数据库定义和数据装入功能,提供数据操纵(包括检索与数据存取操作),数据控制(包括安全性、完整性和并发控制)和数据库维护(包括数据库整理、修改与重定义等)等功能。

DBMS提供应用程序与数据库的接口,允许用户逻辑地访问数据库中的数据,负责逻辑数据与物理地址间的映射,是控制和管理数据库运行的工具。

3. 支持数据库运行的硬、软件环境

(1) 硬件

硬件是指数据库赖以存在的物理设备,包括 CPU、存储器和其他外部设备。数据库系统要求很大的内存,用来存放系统程序、应用程序和表缓冲区。数据库系统对外部存储器(硬盘)有特殊要求,应配置较大容量的磁盘设备,用来存放要长期保存和反复使用的大量数据。磁盘是一种能支持直接数据存取的外部设备,它为索引文件、链接文件、直接存取文件和倒排文件等文件组织形式和数据存取方法提供物理支持,是数据库系统不可缺少的设备。

(2) 软件

数据库系统是一个软件应用系统。计算机应用系统中,最基本的软件是操作系统,其他任何软件系统(包括 DBMS)都必须在它的支持下工作。因此当选某种 DBMS 时,必须选择能对 DBMS 提供支持的操作系统,如要处理汉字,必须配上汉字系统平台。

4. 应用程序

数据库为多用户共享使用,不同用户有不同业务要求,其所需要的数据,已由数据库设计者精心规划在数据库中。如何使用库中数据是用户的事,但其操作权限仅是数据库的一个子集。

5. 数据库管理员 DBA (DataBase Administrator)

在数据库系统中,人的因素起重要作用,应该把 DBA 看成系统不可忽略的一部分。

大型数据库系统,要求配备专职 DBA,他们的职责是对数据库进行管理和维护,保持数据库始终处于最佳工作状态。

数据库的建立,是由计算机专业人员和有关业务人员共同组成的班子,从需求分析开始,汇总所有用户的数据,进行分析、综合、提炼,画出全局逻辑结构图,搞清数据来龙去脉,定义各种数据,包括数据类型、结构、值域、来源、用途、使用权限、保密级别等,最后构成数据字典。这个数据字典供系统参考使用。

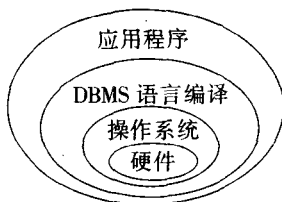


图 1-6 数据库系统

综上所述,数据库系统以硬件为基础,如图 1-6 所示,在硬件基础上层层套上各种软件。外层依赖于内层支持,且可以调用内层但不必考虑内层细节。图中显示了 DBMS 在操作系统支持下与其他语言处理程序可以在一个系统中并存,用户可以根据自己的需要选用 DBMS 或其他语言开发程序。但一定要注意,如果应用程序访问数据库中的数据,就必须通过 DBMS 才能进行。

1.3.2 数据库系统的特点

数据库是个复杂的系统,难以用简练的语言概括其全部特征,实际上关于什么是数据库,至今还没有一个公认的严格定义,当人们从不同角度描述数据库时,可能给出的定义出入很大。

例如, C.J. Date 在他的《数据库系统引论》一书中,从以下三方面描述了数据库。

① 数据库是存储在磁盘或其他存储介质上的数据集合。

② 有若干应用程序以数据库为背景进行检索、修改、插入或删除等操作,还可能有一些联机远程终端用户访问数据库。

③ 数据库是集成的,包含许多用户的数据,每个用户只享用其中一部分数据,不同用

户所使用的数据可以重叠,并且同一片数据可以为多用户共享。

J. Martin 在《数据库组织》一书中对数据库是如下定义的:

数据库是存储在一起的相关数据的集合,这些数据没有不必要的冗余,能为多种应用服务;数据的存储独立于程序;对数据库的插入、修改和检索均能按一种公用的和可控的方法进行;若在一个系统中,存在着结构上完全分离的多个数据表,则称该系统为一个数据库集合。

如上所述,对什么是数据库,从不同角度描述可能差别很大,但是对数据库应具有的基本特点,认识大体上是一致的。我们也应从它的基本特点去认识数据库,下面分析一下数据库究竟有哪些特点。

1. 数据独立

数据独立是数据库方式管理数据的基本优点,也是数据库系统努力追求的目标。数据独立的含义是数据的存储独立于应用程序,以保证数据的改变不致或很少影响应用程序。

数据独立又有两个含义——物理数据独立性和逻辑数据独立性。

物理数据独立性是指数据库物理结构(包括数据的组织和存储、存取方法、外部存储设备等)发生改变时,不会影响到数据库的逻辑结构,而应用程序使用的是逻辑数据,因此可以不必改动应用程序。

逻辑数据独立性是指由于某种原因使数据库的全局逻辑结构发生改变时,用户的应用程序不需要改动,似乎数据库并未发生变化一样。这是因为应用程序是根据该用户的数据视图编写的,仅是全局逻辑数据的一个子集,全局结构变化与否与具体用户无关,只要能从全局导出用户的局部视图就行。

以上两种数据独立性,是由 DBMS 所提供的映象功能实现的。

2. 数据共享

数据共享是指不同用户可以使用同一个数据库,各自取出他们所需要的子集,而且允许子集任意重叠,这是由于数据集成带来的好处。不过当多个用户碰巧同时对某一数据更新时,要进行控制,避免数据发生矛盾,一般 DBMS 都具有并发控制功能。

3. 减少了数据冗余

数据库方式,数据是集成化的,应用程序使用的逻辑文件是取自整体的子集,尽管一个数据可能出现在不同的逻辑文件中,但实际上的物理存储只可能出现一次,这就减少了数据冗余。

然而数据冗余有时难以避免,甚至有时为了某种需要有意地重复存储数据而带来其他方便,所以只说减少数据冗余而不提“避免”二字。冗余数据要占用空间,有时减少了冗余,但需要数据时要利用 CPU 运算从其他数据中推算过来而影响速度,所以有时会采用以空间换速度,适当的冗余可大大减少运算量,提高处理速度。

4. 数据的安全性与完整性

安全性:指数据保密,为防止数据被窃或破坏,DBMS 设置严密措施,层层防护。

完整性:指数据的正确性和一致性。

实现安全性和完整性保护,不同的 DBMS 有不同的方法,功能有强有弱。提升数据安全性和完整性要以 CPU 的时间开销为代价,会使数据库系统的处理速度下降。