



面向“十二五”高职高专规划教材·计算机系列

# SQL Server

## 数据库应用教程

■ 刘甫迎 主编  
■ 李飞跃 向 勇 刘 焱 编著



清华大学出版社 · 北京交通大学出版社

## 内 容 简 介

本书共有 11 章和两个附录,介绍了数据库基础,数据库的由来和发展,数据模型,关系数据库,函数依赖,范式,PD CASE 工具,数据库的设计与维护(CDM、PDM 数据模型),安全性与完整性,并发控制与恢复等数据库及数据库设计的基本原理和技术。详述了后端大型数据库 SQL Server,并将其贯穿全书,特别是其 T-SQL 语言、常用工具、服务器监视、性能优化、基于 ASP.NET+C#的 ADO.NET 数据库访问技术等内容,使本书同时又成了一本学习 SQL Server 2005、SQL Server 2008 应用编程和提高数据库管理员能力的教科书。

本书理论与实践相结合。配有综合应用开发实例、实验指导书、习题、教学大纲,便于学习与教学。可作为高等院校计算机类专业及软件专业的教材,也适于从事数据库应用和开发的人员参考。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13501256678 13801310933

### 图书在版编目(CIP)数据

SQL Server 数据库应用教程 / 刘甫迎主编. —北京:清华大学出版社;北京交通大学出版社, 2010.9

(面向“十二五”高职高专规划教材·计算机系列)

ISBN 978-7-5121-0333-7

I. ①S… II. ①刘… III. ①关系数据库-数据库管理系统, SQL Server-高等学校:技术学校-教材 IV. ①TP311.138

中国版本图书馆 CIP 数据核字(2010)第 172604 号

责任编辑:谭文芳 特邀编辑:宋林静

出版发行:清华大学出版社 邮编:100084 电话:010-62776969 <http://www.tup.com.cn>

北京交通大学出版社 邮编:100044 电话:010-51686414 <http://press.bjtu.edu.cn>

印刷者:北京交大印刷厂

经 销:全国新华书店

开 本:185×260 印张:22.75 字数:579 千字

版 次:2010 年 9 月第 1 版 2010 年 9 月第 1 次印刷

书 号:ISBN 978-7-5121-0333-7/TP·619

印 数:1~4 000 册 定价:37.00 元

---

本书如有质量问题,请向北京交通大学出版社质监组反映。对您的意见和批评,我们表示欢迎和感谢。

投诉电话:010-51686043, 51686008; 传真:010-62225406; E-mail: [press@bjtu.edu.cn](mailto:press@bjtu.edu.cn)。

# 前 言

20 世纪 60 年代诞生的数据库技术已成为计算机领域中最重要技术之一，是 21 世纪信息时代用于各行各业不可或缺的“武器”。其建设规模和使用水平已成为衡量一个国家信息化程度的重要标志，越来越多的人希望学习数据库方面的知识。

美国微软公司的 SQL Server 数据库管理系统是世界上最快的关系数据库之一，其价廉、功能强大和用户广泛对大型数据库管理系统的工业标准 Oracle 提出了严峻挑战。“数据库原理及应用（SQL Server）”课程已经成为高等学校计算机类专业和信息管理类专业等专业普遍设置的一门专业基础课。

本书的主要特点是：

1. 理论与实践相结合，既有实用性，又有可持续性。本书既有许多 SQL Server 教材所缺乏的数据库的体系结构（见第 1 章），数据模型、关系代数、关系演算（见第 2 章），函数依赖、范式（见第 5 章），安全性与完整性、并发控制、数据库后备和恢复（见第 9 章）等数据库的基本原理和技术理论，又注重与实践相结合，突出实际动手能力的培养。配有实例、案例（见第 11 章）、实验指导书、专周（见附录 B）等，以便于读者参考、使用。

2. 本书着眼于 SQL Server 2008 版本的新特点及 SSMS 增强，同时也介绍了 SQL Server 2005。既体现了其最新技术的先进性，又注意兼顾 2005 版本用户较多的可用性。

3. 增加了 PowerDesigner CASE 技术，进行 CDM、PDM 建模来设计数据库（见第 6、7 章），这是其他 SQL Server 教材所没有的体现数据库设计方向的技术。

4. 以职业竞争力为导向，实行岗位职能教育。本书突出数据库管理员岗位职能教育（见第 9 章等），又辅以 SQL Server 应用编程的岗位职能教育（见第 8、10、11 章等），力图使学生学习本书后便基本可以编制 SQL Server 应用程序、设计和管理数据库系统。

5. 本书的 SQL Server 应用程序编程使用的是基于 .NET 平台（ASP.NET+C# + ADO.NET）进行 B/S 模式编程的主流技术（见第 10、11 章），而不是许多 SQL Server 教材介绍的传统的基于 VB 平台进行 C/S 模式编程的技术。

6. 本书将实体-联系模型（E-R Model）单独放在一章（见第 3 章），还增加了其他 SQL Server 教材所少有的“并发控制”（见第 9 章）和“基于 ADO.NET 的 SQL Server 数据库访问技术”（见第 10 章）等内容，使本书更具系统性、全面性。

7. 本书内容安排按人们学习认知的规律，即从简单到复杂的认识转化过程与基于工作的过程相结合。本书以使用 SQL Server 数据库的 FoxERP\_HRSA 人事系统作为项目驱动（见第 11 章），经历了建库和编程从理论（见第 1、3 章、2.1~2.4 节）到实践（见 2.5 节、第 4 章），从实践到理论（见第 5、6 章），再到实践的过程（见第 7、8、10、11 章），符合人们学习的认知规律。

8. 本书附有教学大纲（见附录 A）、习题，图文并茂，便于学习与教学。

9. 本书作者长期从事数据库教材的编写和教学、科研工作，对此课程有丰富的教学经

验，对编写此书起到很好的作用。

本书由刘甫迎主编，李飞跃、向勇、刘焱编著。刘甫迎编著第1章、第3章、第5章、第10章和附录A；李飞跃编著第2.5节（除2.5.4、2.5.5外）、第4章（除4.2、4.3.2节外）、第7.5节（除7.5.1、7.5.4节外）、第9章（除9.1.1、9.1.3、9.1.6、9.3.3、9.6、9.7节外）、附录B；向勇编著第2章（除2.5.1~2.5.3节外）、第7章（除7.5.2、7.5.3节外）、第8章；刘焱编著第6章、第9.1.3、9.1.6、9.3.3节及9.6、9.7节、第11章。全书由刘甫迎教授统稿。在本书编著和出版的过程中出版社的编辑做了不少工作，给予了很大的帮助，学生李朝蓉等做了一些辅助工作，在此一并表示感谢！

由于水平有限，错误难免，欢迎批评指正。

编 者

# 目 录

|                                           |    |
|-------------------------------------------|----|
| 第1章 数据库导论 .....                           | 1  |
| 1.1 数据库的基本概念 .....                        | 1  |
| 1.1.1 信息处理及数据处理 .....                     | 1  |
| 1.1.2 数据模型和数据库技术 .....                    | 2  |
| 1.2 数据库的由来和发展 .....                       | 2  |
| 1.2.1 人工管理阶段 .....                        | 3  |
| 1.2.2 文件系统阶段 .....                        | 4  |
| 1.2.3 数据库阶段 .....                         | 4  |
| 1.2.4 高级数据库阶段 .....                       | 6  |
| 1.3 数据库的三级体系结构 .....                      | 10 |
| 1.4 数据库系统和数据库管理员 .....                    | 14 |
| 1.5 数据库管理系统 .....                         | 17 |
| 习题 .....                                  | 20 |
| 第2章 关系模型及 SQL Server 数据库管理系统 .....        | 21 |
| 2.1 关系模型和基本概念 .....                       | 21 |
| 2.1.1 关系的定义 .....                         | 21 |
| 2.1.2 关系模型 .....                          | 22 |
| 2.1.3 关系数据语言概述 .....                      | 23 |
| 2.2 关系代数 .....                            | 24 |
| 2.2.1 传统的集合运算 .....                       | 24 |
| 2.2.2 专门的关系运算 .....                       | 24 |
| 2.3 关系演算 .....                            | 29 |
| 2.3.1 元组关系演算 .....                        | 29 |
| 2.3.2 域关系演算 .....                         | 31 |
| 2.4 关系数据库标准语言——SQL .....                  | 31 |
| 2.5 SQL Server 关系数据库管理系统 .....            | 34 |
| 2.5.1 SQL Server 概述 .....                 | 34 |
| 2.5.2 安装、测试与卸载 .....                      | 36 |
| 2.5.3 SQL Server 常用工具 .....               | 38 |
| 2.5.4 SQL Server 服务器管理 .....              | 47 |
| 2.5.5 SQL Server 2008 的新特点及 SSMS 增强 ..... | 56 |
| 习题 .....                                  | 62 |
| 第3章 实体-联系模型 .....                         | 64 |

|              |                               |           |
|--------------|-------------------------------|-----------|
| 3.1          | 实体和实体集合 .....                 | 64        |
| 3.2          | 联系和联系集合 .....                 | 65        |
| 3.3          | 属性, 映射限制和关键字 .....            | 66        |
| 3.3.1        | 属性 .....                      | 66        |
| 3.3.2        | 映射限制 .....                    | 66        |
| 3.3.3        | 关键字 .....                     | 67        |
| 3.4          | 实体联系图解和将其归纳为表 .....           | 69        |
| 3.4.1        | 实体联系图 .....                   | 69        |
| 3.4.2        | 把 E-R 图转变为表 .....             | 71        |
| 3.5          | 概括和聚集 .....                   | 73        |
| 3.5.1        | 概括 .....                      | 73        |
| 3.5.2        | 聚集 .....                      | 74        |
| 3.6          | E-R 数据库模式文件设计 .....           | 75        |
| 3.6.1        | 映射基本集 .....                   | 75        |
| 3.6.2        | 实体集和联系集的使用 .....              | 76        |
| 3.6.3        | 扩展 E-R 特征的使用 .....            | 76        |
|              | 习题 .....                      | 77        |
| <b>第 4 章</b> | <b>SQL Server 的 SQL .....</b> | <b>78</b> |
| 4.1          | SQL Server 的数据类型 .....        | 78        |
| 4.2          | SQL 中的函数和表达式 .....            | 81        |
| 4.2.1        | SQL 语言中的函数 .....              | 81        |
| 4.2.2        | SQL 语言中的运算符 .....             | 83        |
| 4.2.3        | SQL 语句中的子句表达式 .....           | 85        |
| 4.3          | 定义、修改、删除表 .....               | 86        |
| 4.3.1        | 创建基本表 .....                   | 86        |
| 4.3.2        | 约束 .....                      | 87        |
| 4.3.3        | 修改表结构 .....                   | 91        |
| 4.3.4        | 索引的定义 .....                   | 92        |
| 4.3.5        | 删除表和索引 .....                  | 93        |
| 4.4          | SQL 的数据查询 .....               | 93        |
| 4.4.1        | 简单查询 .....                    | 93        |
| 4.4.2        | 连接查询 .....                    | 96        |
| 4.4.3        | 嵌套查询 .....                    | 97        |
| 4.4.4        | 分组计算查询 .....                  | 99        |
| 4.4.5        | 集合的并运算 .....                  | 102       |
| 4.5          | 插入、修改和删除数据操纵语言 .....          | 103       |
| 4.5.1        | 将新行插入表 .....                  | 103       |
| 4.5.2        | 修改表的行 .....                   | 103       |
| 4.5.3        | 从表删除行 .....                   | 104       |

|       |                             |     |
|-------|-----------------------------|-----|
| 4.6   | 视图 .....                    | 104 |
| 4.6.1 | 视图的定义 .....                 | 104 |
| 4.6.2 | 视图的查询 .....                 | 105 |
| 4.6.3 | 视图修改 .....                  | 105 |
|       | 习题 .....                    | 106 |
| 第 5 章 | 关系数据库设计理论 .....             | 108 |
| 5.1   | 引言 .....                    | 108 |
| 5.2   | 函数依赖 .....                  | 109 |
| 5.3   | 范式 .....                    | 111 |
| 5.3.1 | 第 1 范式 .....                | 111 |
| 5.3.2 | 第 2 范式 .....                | 112 |
| 5.3.3 | 第 3 范式 .....                | 112 |
| 5.3.4 | BC 范式 .....                 | 113 |
| 5.4   | 多值依赖和第 4 范式 .....           | 114 |
| 5.5   | 连接依赖和第 5 范式 .....           | 116 |
| 5.5.1 | 连接依赖 .....                  | 116 |
| 5.5.2 | 第 5 范式 .....                | 117 |
|       | 习题 .....                    | 117 |
| 第 6 章 | PowerDesigner CASE 技术 ..... | 119 |
| 6.1   | CASE 工具的概念 .....            | 119 |
| 6.2   | PD 分析设计过程及若干级建模技术 .....     | 122 |
| 6.2.1 | PD 的分析设计过程 .....            | 122 |
| 6.2.2 | PD 的若干级建模功能 .....           | 123 |
| 6.3   | PD 的功能 .....                | 124 |
| 6.3.1 | PD 的一般功能 .....              | 124 |
| 6.3.2 | PD 12.5 的主要特征 .....         | 126 |
| 6.4   | PD 的分析设计环境 .....            | 126 |
| 6.4.1 | PD 的环境要求及安装 .....           | 126 |
| 6.4.2 | PD 主界面 .....                | 128 |
| 6.4.3 | PD 的分析设计环境 .....            | 130 |
| 6.4.4 | PD 的公共资源 .....              | 132 |
|       | 习题 .....                    | 132 |
| 第 7 章 | 数据库的设计与维护 .....             | 134 |
| 7.1   | 数据库设计概述 .....               | 134 |
| 7.1.1 | 数据库工程、生存期及其他 .....          | 134 |
| 7.1.2 | 数据库设计的需求分析 .....            | 136 |
| 7.2   | 用 PD 创建 CDM .....           | 138 |
| 7.2.1 | 概念数据模型的基础 .....             | 138 |
| 7.2.2 | 用 PD 建立 CDM .....           | 140 |

|        |                                      |     |
|--------|--------------------------------------|-----|
| 7.2.3  | 定义业务规则 .....                         | 143 |
| 7.2.4  | 定义 CDM 中的域 .....                     | 145 |
| 7.2.5  | 定义数据项 .....                          | 147 |
| 7.2.6  | 定义实体 .....                           | 150 |
| 7.2.7  | 定义联系 .....                           | 155 |
| 7.2.8  | 定义继承 .....                           | 160 |
| 7.2.9  | 定义检查约束参数 .....                       | 162 |
| 7.2.10 | 由 CDM 生成 PDM .....                   | 163 |
| 7.3    | 用 PD 创建物理数据模型 .....                  | 166 |
| 7.3.1  | PDM 及其与 DBMS 的关系 .....               | 166 |
| 7.3.2  | 建立物理数据模型 .....                       | 166 |
| 7.3.3  | 定义索引 .....                           | 169 |
| 7.3.4  | 定义视图 .....                           | 171 |
| 7.3.5  | PDM 中的用户管理 .....                     | 172 |
| 7.3.6  | 检查 PDM 中的对象 .....                    | 172 |
| 7.3.7  | 逆向工程 .....                           | 173 |
| 7.4    | 数据库的实现 .....                         | 175 |
| 7.4.1  | 数据库的实现与运行概述 .....                    | 175 |
| 7.4.2  | 由 PDM 生成数据库 SQL 脚本 .....             | 177 |
| 7.5    | 用 SQL Server 创建和管理数据库 .....          | 178 |
| 7.5.1  | SQL Server 数据库的组成 .....              | 178 |
| 7.5.2  | 数据库的创建及管理 .....                      | 181 |
| 7.5.3  | 在图形界面下创建及管理表 .....                   | 186 |
| 7.5.4  | 在图形界面下创建及管理视图 .....                  | 190 |
|        | 习题 .....                             | 193 |
| 第 8 章  | SQL Server 的 Transact-SQL 程序设计 ..... | 195 |
| 8.1    | Transact-SQL 语言基础 .....              | 195 |
| 8.1.1  | 语法规则 .....                           | 195 |
| 8.1.2  | 常量和变量 .....                          | 198 |
| 8.1.3  | 批和脚本 .....                           | 201 |
| 8.2    | 结构化流程控制 .....                        | 202 |
| 8.2.1  | 顺序结构与 GOTO .....                     | 202 |
| 8.2.2  | 分支结构 .....                           | 202 |
| 8.2.3  | 循环结构 .....                           | 204 |
| 8.3    | 其他函数 .....                           | 206 |
| 8.3.1  | 其他常用内置函数 .....                       | 206 |
| 8.3.2  | 用户自定义函数 .....                        | 207 |
| 8.4    | 游标 .....                             | 211 |
| 8.4.1  | 游标的概念 .....                          | 211 |

|        |                                           |     |
|--------|-------------------------------------------|-----|
| 8.4.2  | 游标数据的操作 .....                             | 214 |
| 8.5    | 存储过程 .....                                | 215 |
| 8.5.1  | 存储过程的概念 .....                             | 215 |
| 8.5.2  | 创建与执行存储过程 .....                           | 217 |
| 8.5.3  | 维护存储过程 .....                              | 220 |
| 8.5.4  | 数据库的存储过程设计实例 .....                        | 222 |
|        | 习题 .....                                  | 223 |
| 第 9 章  | 数据库的保护与维护 .....                           | 224 |
| 9.1    | 数据库的安全性 .....                             | 224 |
| 9.1.1  | 数据库的安全体系及存取机制 .....                       | 224 |
| 9.1.2  | SQL Server 的安全认证模式及登录账户 .....             | 228 |
| 9.1.3  | 用户的建立和管理 .....                            | 232 |
| 9.1.4  | 特权 .....                                  | 235 |
| 9.1.5  | 角色 .....                                  | 240 |
| 9.1.6  | 证书及数据加密 .....                             | 244 |
| 9.2    | 数据完整性 .....                               | 248 |
| 9.2.1  | 完整性约束 .....                               | 249 |
| 9.2.2  | 数据库触发器 .....                              | 251 |
| 9.3    | 并发控制 .....                                | 255 |
| 9.3.1  | 数据库不一致的类型 .....                           | 255 |
| 9.3.2  | 封锁 .....                                  | 256 |
| 9.3.3  | 事务 .....                                  | 258 |
| 9.4    | 数据库后备与恢复 .....                            | 260 |
| 9.4.1  | 数据日志 .....                                | 260 |
| 9.4.2  | 数据库后备 .....                               | 262 |
| 9.4.3  | 数据库恢复 .....                               | 266 |
| 9.5    | 数据传输 .....                                | 270 |
| 9.6    | SQL Server 服务器监视 .....                    | 273 |
| 9.6.1  | 服务器的监视手段 .....                            | 274 |
| 9.6.2  | 使用 SQL Server 事件探查器 .....                 | 274 |
| 9.7    | SQL Server 数据库的性能优化 .....                 | 279 |
|        | 习题 .....                                  | 281 |
| 第 10 章 | 基于 ASP.NET 应用程序的 SQL Server 数据库访问技术 ..... | 283 |
| 10.1   | Web 程序设计 .....                            | 283 |
| 10.1.1 | B/S 模式编程及 .NET 平台 .....                   | 283 |
| 10.1.2 | IIS 对 Web 程序的支持 .....                     | 290 |
| 10.2   | ADO.NET 简介 .....                          | 292 |
| 10.2.1 | ADO.NET 的对象模型 .....                       | 292 |
| 10.2.2 | 常用的 ADO.NET 类的属性和方法 .....                 | 293 |

|        |                                      |     |
|--------|--------------------------------------|-----|
| 10.3   | ASP.NET 应用程序访问 SQL Server 数据库 .....  | 295 |
| 10.3.1 | 添加 ASP.NET 用户并编辑其对数据库的访问权限 .....     | 295 |
| 10.3.2 | ASP.NET 应用程序连接数据库 .....              | 300 |
| 10.3.3 | 在 Web Form 网页中显示数据 .....             | 302 |
| 10.4   | ASP.NET 数据源控件及数据绑定控件 .....           | 304 |
| 10.4.1 | 绑定到数据库 .....                         | 304 |
| 10.4.2 | 绑定到对象 .....                          | 306 |
| 10.4.3 | 数据绑定控件 .....                         | 308 |
|        | 习题 .....                             | 309 |
| 第 11 章 | SQL Server 应用开发实例——企业资源计划的人事系统 ..... | 310 |
| 11.1   | FoxERP 企业资源计划系统的架构、平台及特点 .....       | 310 |
| 11.1.1 | FoxERP 的组成及 ASP.NET+C#开发环境及功能 .....  | 310 |
| 11.1.2 | FoxERP 的系统架构 .....                   | 311 |
| 11.1.3 | FoxERP-HRSA 人事管理子系统 .....            | 311 |
| 11.2   | 系统分析与设计 .....                        | 312 |
| 11.2.1 | 概述 .....                             | 312 |
| 11.2.2 | 概念数据模型 .....                         | 315 |
| 11.2.3 | 物理数据模型 .....                         | 316 |
| 11.2.4 | 面向对象的模型 OOM .....                    | 316 |
| 11.3   | 数据库表的结构 .....                        | 319 |
| 11.3.1 | 表清单 .....                            | 319 |
| 11.3.2 | 表中字段及相关设置 .....                      | 320 |
| 11.4   | 主要关键技术 .....                         | 326 |
| 11.4.1 | 系统逻辑层功能的实现 .....                     | 326 |
| 11.4.2 | 安全性的设计 .....                         | 330 |
| 11.4.3 | 母版和 CSS .....                        | 332 |
| 11.5   | 用户使用说明书 .....                        | 334 |
| 11.5.1 | 系统使用平台 .....                         | 334 |
| 11.5.2 | 登录和进入系统 .....                        | 337 |
| 11.5.3 | FoxERP-HRSA 人事管理子系统的使用 .....         | 338 |
| 11.6   | 部分源程序 .....                          | 339 |
| 11.6.1 | 存储过程代码 .....                         | 339 |
| 11.6.2 | 数据访问层代码 .....                        | 342 |
| 11.6.3 | 业务逻辑层代码 .....                        | 343 |
|        | 习题 .....                             | 345 |
| 附录 A   | 《SQL Server 数据库原理及应用》教学大纲 .....      | 346 |
| 附录 B   | 实验指导书 .....                          | 348 |
| 参考文献   | .....                                | 351 |

# 第 1 章 数据库导论

本章介绍数据库的意义、数据库的发展史、数据库的体系结构、数据库系统及数据库管理系统等，以便使读者对数据库的概念有一个基本的了解。

## 1.1 数据库的基本概念

### 1.1.1 信息处理及数据处理

诞生于 20 世纪中叶的计算机科学较之其他现代科学技术的发展更加迅速，尤其是在 21 世纪其发展速度更加迅猛。众所周知，21 世纪是信息时代，如何组织和利用那些庞大的信息和知识已成为衡量一个国家科学技术水平高低的重要标志。

早在 20 世纪 60 年代，数据库技术作为现代信息系统的一门基础学科便应运而生了。现在，数据库技术已成为计算机领域中最重要技术之一，它是软件学科中一个独立的分支。它的出现使得计算机应用渗透到工农业生产、商业、行政、教育、科学研究、工程技术和国防军事等领域。管理信息系统（Management Information System, MIS）、办公自动化系统（Office Automation, OA）、决策支持系统等都是使用了数据库管理系统或数据库技术的计算机应用系统。

数据库（DataBase, DB）是存储在一起的相关数据的集合，是存储数据的“仓库”。因此，要理解数据库就需要先了解在数据处理领域中常遇到的两个基本概念：“信息”（Information）和“数据”（Data）。

信息是关于现实世界事物的存在方式或运动状态反映的组合。例如，上课用的黑板，其颜色是黑的，形状是矩形，尺寸是长 3.2 米，高 1.4 米，材料是木材，这些都是关于黑板的信息，都是关于黑板的存在状态的反映，从不同角度“反映”或“刻画”了黑板这个事物。信息源于物质和能量，一切事物，包括自然和人类都产生信息，信息是物质和能量形态的反映，它不可能脱离物质而存在。信息传递需要物质载体，信息的获取和传递要消耗能量。信息是可以感知的，信息可以存储，并且可以加工、传递和再生。电子计算机是信息处理领域中最先进的工具之一，人类对收集到的信息可以进行取舍整理。几乎和信息同样广泛使用的另一个概念是“数据”。所谓数据，通常指用符号记录下来的可加以鉴别的信息。例如，为了描述黑板的信息，可以用一组数据“黑色、矩形、3.2 米×1.4 米”来表示，由于“黑色”、“矩形”、“3”、“米”等符号已经被人们赋予了特定的语义，所以它们就具有了传递信息的功能。

从上面的例子中可以看到信息和数据之间的固有联系：数据是信息的符号表示或称为载体，信息则是数据的内涵，是对数据的语义解释。但另一方面，某一具体信息与表示它的数据的这种对应关系又因环境而异。同一信息可能有不同的符号表示，同一数据也可能有不同的解释。数据处理领域中的数据概念较之科学计算领域中数据概念已经大大地拓宽了。定

义中所说的符号, 不仅包含数字符号, 而且包含文字、图像和其他符号; 而所谓“记录下来”也不仅是指用笔写在纸上, 它还包括磁记录、光刻记录等各种记录形式。

由于信息是现实世界中事物的存在方式和运动状态的反映, 而现实世界的事物常常是相互关联的, 这就使得人们在了解、掌握事物之间的固有联系和运动规律的基础上, 可以从一些已知的信息出发, 经过演绎推理, 导出新的信息, 为人类社会生活的各种需要服务, 这就是常说的“信息处理”。例如, 根据上述黑板的尺寸数据和木工定额标准, 可以推算出制作黑板所需的木材数量和制作费用。

一般情况下, 人们将原始信息表示成源数据, 然后对这些源数据进行综合推导加工, 得出新的数据。这些结果数据表示了新的信息, 可以作为某种决策的依据 (或用于新的推导加工)。其整个过程通常称为“数据处理”。

电子计算机使大规模的数据处理成为可能, 它和通信、网络技术的发展一起, 进一步推动了信息处理和信息利用的社会化, 极大地增强了人类社会处理信息的能力。

### 1.1.2 数据模型和数据库技术

数据库系统就是实现有组织地、动态地存储大量相关数据, 方便用户访问的计算机软硬件资源组成的系统。而数据库技术是研究数据库的结构、存储、设计和使用的一门软件学科。因此, 数据库技术主要是研究如何存储、使用和管理数据。在计算机应用中, 数据处理占的比重最大, 而数据库系统是数据处理的核心机构, 所以它的效能往往决定了整个计算机应用的经济效益。

数据库离不开数据模型。数据模型是对现实世界客观事物及其联系的描述, 它反映数据项之间和记录之间的联系, 在数据库技术中使用模型的概念描述数据库的结构与语义。常用的 3 种数据模型是: 层次模型 (Hierarchical Mode)、网状模型 (Network Model) 和关系模型 (Relational Model)。此外, 还有面向对象模型 (Object-Oriented Model) 等 (关系数据模型将在第 2 章详细叙述)。

数据库这门学科与其他基础软件、系统软件、应用软件有着密切的联系。例如, 操作系统, 数据库技术是在操作系统的文件系统的基础上发展起来的, 而且数据库系统本身就是在操作系统 (例如, Windows) 的支持下才能工作。数据库与网络技术和多媒体技术的关系也很密切, 例如, 分布式数据库要使用网络, 甚至可在国际互联网 (Internet) 上交换海外的数据、声音、图像、图片等多媒体信息。数据库技术与数据结构的关系也是不密不可分的, 数据库技术不仅要用到数据结构的知识, 而且丰富了数据结构的内容。程序设计是使用数据库系统的最基本的方式, 因为数据库中大量的应用程序过程大都是用高级语言加上数据库的操纵语言编写的。集合论、数理逻辑是关系数据库的理论基础, 其很多概念、术语、思想都直接用到关系数据库中。

## 1.2 数据库的由来和发展

数据库这个名词起源于 20 世纪 50 年代, 当时美国为了战争的需要, 把各种情报集中在一起, 存入计算机, 称为 Information Base 或 DataBase。1963 年美国 Honeywell 公司的 IDS (Integrated Data Store, 集成数据存储) 系统投入运行, 揭开了数据库技术的序幕。

1965 年美国利用数据库设计了阿波罗登月火箭, 推动了数据库技术的产生。当时社会上产生了许多种 DataBase 或 DataBank, 但基本上都是文件系统的扩充。1968 年美国 IBM 公司推出了层次模型的 IMS 数据库系统, 并于 1969 年形成产品; 1969 年, 推出了 COBOL 语言的 CODASL (Conference on Data System Language, 数据系统语言协会) 组织的数据库任务组 (DBTG) 发表了网状数据库系统的标准文本 (1971 年正式通过); 1970 年年初, IBM 公司的高级研究员 E.F.Codd 发表论文提出了关系模型, 奠定了关系数据库的理论基础。

20 世纪 70 年代是数据库蓬勃发展的年代。网状系统和层次系统占领了市场, 关系系统开始处于实验阶段, IBM 公司研制出了原型关系语言 System R。1979 年关系软件 (Relational Software) 公司推出了第一个基于 SQL 的商用关系数据库产品 ORACLE。

从 20 世纪 80 年代起, 关系数据库产品已相当成熟, 取代了网状系统和层次系统的市场。同时关系数据库理论也日趋完善, 走向更高级的阶段, 有了分布式数据库系统 (Distributed DataBase System, DDBS) 等。后来, 从不同的计算机应用领域提出了许多数据库的非传统应用课题, 诸如多媒体数据、空间数据、时序数据、科学数据、复杂对象、知识、超文本管理等。为了适应这类应用的需要, 提出了不少新的概念、新的数据模型和系统结构。经过几年的研究和实践, 逐步形成了面向对象数据库系统 (Object Oriented DataBase System, OODBS)、主动数据库系统 (Active DataBase Systems, ADBS)、大型知识库系统 (Large Knowledge Base Systems, LKBS)、数据库中的知识发现 (Knowledge Discovery in DataBase, KDD) 及科学数据库 (Science DataBase, SDB) 等热点, 为 21 世纪数据库技术蓬勃发展奠定了基础。了解数据库的由来和发展的历史及数据管理技术各阶段的特点, 对学习好数据库十分必要。

综观数据管理技术的发展可知: 它与硬件 (主要是外部存储器)、软件及计算机应用的范围有密切关系。数据管理技术大致经过以下 4 个阶段: 人工管理阶段、文件系统阶段、数据库阶段及高级数据库阶段。

### 1.2.1 人工管理阶段

人工管理阶段 (20 世纪 50 年代中期以前) 的计算机主要用于科学计算。硬件中的外存只有卡片、纸带等。软件只有汇编语言, 没有数据管理方面的软件。数据处理的方式基本上是批处理。这个时期的数据管理具有如下特点。

#### 1) 数据不保存

进行某一课题计算时将原始数据随程序一起输入主存, 运算处理后将结果数据输出。任务完成后, 数据空间同程序空间一起释放。

#### 2) 没有专用软件对数据进行管理

每个应用程序要包括存储结构、存取方法、输入/输出方式等, 数据结构与程序不具有独立性, 一旦存储结构改变, 就必须由应用程序员修改程序。由于程序直接面向存储结构, 因此不存在逻辑结构与物理结构的区别。

#### 3) 只有程序 (Program) 的概念、没有文件 (File) 的概念

即使有文件, 也大多是顺序文件, 其他组织方式必须由程序员自行设计与安排。

#### 4) 数据面向应用

数据面向应用即一组数据对应于一个程序。由于各应用程序处理的数据不会毫无联系, 程序之间会有重复。

### 5) 对数据的存取以记录为单位

由于对数据的存取是以记录为单位, 因此其灵活性差。

## 1.2.2 文件系统阶段

文件系统阶段(20 世纪 50 年代后期至 60 年代中后期)的计算机不仅用于科学计算, 还大量用于信息管理。外存已有磁盘、磁鼓等直接存取存储设备。软件方面出现了高级语言和操作系统。操作系统中的文件系统(有时也称为“信息处理模块”)是专门处理外存的数据管理软件。处理数据方式有批处理, 也有联机实时处理。这一阶段数据管理情况如下所述。

### 1) 特点

(1) 数据可长期保存在外存的磁盘上。用户可以随时通过程序对文件进行查询、修改及删除等处理。由于计算转向管理, 数据处理的工作量增大。

(2) 数据的物理结构与逻辑结构有了区别, 但较简单。程序与设备之间有设备独立性(程序只需用文件名与数据打交道, 不必关心数据的物理位置), 由操作系统的文件系统提供存取方法(读/写)。由存取方法实现数据的逻辑结构与物理结构之间的转换。

(3) 文件的形式已多样化, 有索引文件、链接文件和直接存取等, 因而对文件的记录可顺序访问, 也可随机访问。但文件之间是独立的, 联系要通过程序去构造, 文件的共享性差。

(4) 有了存储文件以后, 数据不再仅仅属于某个特定的程序, 而可以重复使用。但文件结构的设计仍然是基于特定的用途, 程序仍然是基于特定的物理结构和存取方法编制的, 因此, 数据结构与程序之间的依赖关系并未根本改变。

(5) 对数据的存取基本上还是以记录为单位。

### 2) 缺陷

在文件系统中, 改变存储设备, 不必改变应用程序。虽然文件系统提供了存取方法, 但这只是初级的数据管理。这种文件系统, 还未能彻底体现用户观点下的数据逻辑结构独立于数据在外存的物理结构要求。因此, 数据的物理结构修改时, 仍然需要修改用户的应用程序。

文件系统有以下 3 大缺陷:

(1) 数据冗余性(Redundancy), 由于文件之间缺乏联系, 造成每个应用程序都有对应的文件, 有可能同样的数据在多个文件中重复存储;

(2) 不一致性(Inconsistency), 这往往是由数据冗余造成的, 在进行更新操作时, 稍不谨慎, 就可能使同样的数据在不同的文件中不一样;

(3) 数据联系弱(Poor Data Relationship), 这是文件之间独立, 缺乏联系造成的。

由于上述原因, 促使人们开始研究一种新的数据管理技术, 这就是 20 世纪 60 年代末产生的数据库技术。

## 1.2.3 数据库阶段

数据库阶段(20 世纪 60 年代末开始), 即 20 世纪 60 年代末, 磁盘技术取得了重大进展, 大容量(数百兆字节以上)和快速存取的磁盘陆续进入市场, 成本有了很大的下降, 为数据库技术的实现提供了物质条件。

在 60 年代中期, 出现的大多数系统(DataBase 或 DataBank)还不能称为真正的数据库系统。数据管理技术进入数据库阶段的标志是前面讲到的 60 年代后期的 3 大事件: 1969

年 IBM 公司推出的 IMS 产品 (Information Management System, 层次数据库系统) 和 CODASL 研究和建议的 DBTG 系统 (O 网状数据库系统), 及 1970 年起, IBM 公司的 E.F.Codd 连续发表一系列论文, 奠定了关系数据库理论基础。

70 年代以来, 数据库技术得到迅速发展, 并投入使用。数据库系统阶段根据其支持的数据模型 (Data Model), 到目前为止经历了 3 个时代: 从上面讲到的层次网状时代到关系模型时代又到面向对象时代。关系模型时代从 70 年代初 E.F.Codd 奠定了关系数据库理论基础后, 70 年代末推出了一些试验系统, 80 年代初出现了一批商品化的关系数据库系统, 如 Oracle、SQL/DS、DB2、INGRES、INFORMIX、UNIFY 及 dBASE、FoxBASE 等。SQL 语言在 1986 年被美国 ANSI 和国际标准化组织 (ISO) 采纳为关系数据库语言的国际标准。

与文件系统相比, 数据库系统克服了文件系统的缺陷, 提供了对数据更高级、更有效的管理。概括起来, 数据库技术的管理方式具有以下特点。

### 1) 采用复杂的数据模型 (结构)

数据模型不仅要描述数据本身的特点, 还要描述数据之间的联系。这种联系是通过存取路径来实现的。通过一切存取路径来表示自然的数据联系是数据库与传统文件的根本区别。这样数据不再面向特定的某个或多个应用, 而是面向整个应用系统。数据冗余明显减少, 实现了数据共享。

### 2) 有较高的数据独立性

在数据库系统中, 系统提供映像的功能, 确保应用程序对数据结构和存取方法有较高的独立性。数据的物理结构和逻辑结构差别可以很大。用户以简单的逻辑结构操作数据而无须考虑数据的物理结构。数据库系统的结构分成用户的逻辑结构、数据库的整体逻辑结构和数据库的物理结构, 如图 1-1 所示。用户 (应用程序或终端用户) 的数据和外存中的数据之间的转换由数据库管理系统实现。为提高效率、减少冗余或增加新的数据, 常需改变数据结构。在改变物理结构时, 不影响整体逻辑结构、用户的逻辑结构及应用程序, 这样就认为数据库达到了物理数据独立性。在改变整体逻辑时, 不影响用户的逻辑结构及应用程序, 这样就认为数据库达到了逻辑数据独立性。

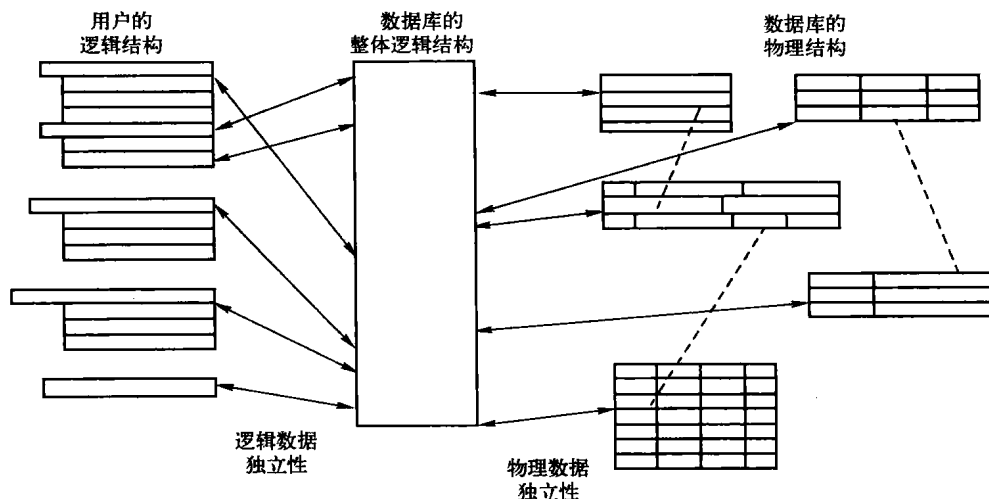


图 1-1 数据库系统的结构

### 3) 数据库系统为用户提供了方便的用户接口

用户可使用查询语言或简单的终端命令操作数据库,也可以使用程序方式(用高级语言如 C、Fortran 等语言和数据库操纵语言编制的程序)操作数据库。

### 4) 提供数据控制功能

(1) 数据完整性: 保证数据库始终包含正确的数据。用户可设计一些完整性规则以确保数据值的正确性。例如, 可把数据值限制在某个范围内, 并对数据值之间的联系进行各种检验。

(2) 数据安全性: 保证数据的安全和机密, 防止数据丢失或被窃取。

(3) 数据库的并发控制: 避免并发程序之间的相互干扰, 防止数据库数据被破坏, 杜绝提供给用户不正确的数据。

(4) 数据的恢复: 在数据库被破坏时或数据不可靠时, 系统有能力把数据库恢复到最近某个时刻的正确状态。

### 5) 可以数据项为单位操作

对数据库的操作除了以记录为单位外, 还可以数据项为单位。

数据库阶段的程序和数据的联系可用图 1-2 表示。

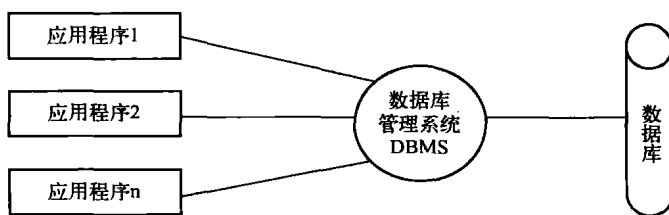


图 1-2 数据库阶段的程序和数据的联系

综上所述, 数据库可以定义为: 一个存储起来为某个特定组织的多种应用服务并具有尽可能小的冗余度的互相关联的数据集合。其数据结构独立于使用数据的程序, 对数据的增添、删除、修改及检索, 由系统进行统一的控制, 而且数据模型也有利于将来应用的发展。

从文件系统发展到数据库技术是信息处理领域的一个重大变化。在文件系统阶段程序设计处于主导地位, 数据只起着服从程序设计需要的作用; 而在数据库方式下, 数据开始占据了中心位置, 数据的结构设计成为信息系统首先关心的问题, 而利用这些程序的应用程序设计则退居到以既定的数据结构为基础的外围地位。

目前在国内外数据库应用已相当普及, 各行业都建立了以数据库技术为基础的大型计算机网络系统, 并在因特网 (Internet) 的基础上建立了国际性联机检索系统, 其应用深入到人类社会生活的各个领域。

## 1.2.4 高级数据库阶段

高级数据库阶段 (20 世纪 70 年代后期开始) 的主要标志是分布式数据库系统、智能数据库系统、面向对象数据库的出现。

## 1. 分布式数据库系统

集中式数据库把数据集中在一个数据库中集中管理,减少了数据冗余和不一致性,而且数据联系比文件系统强得多。但集中式系统也有弱点,如其系统庞大,通信拥挤等。为了真正实现对分布在不同地方的数据资源共享,早在 70 年代就开始了分布式数据库(Distributed DataBase Systems)的研究。经过 10 多年的努力,到 1986 年在软件市场上开始出现了分布式数据库产品。当时,关系技术公司(现为 INGRES 公司)宣布了称为 INGRES/STAR 的分布式版本。其后不久,Oracle 公司也宣布了称为 SQL\*STAR 的 Oracle 分布式版本。这些年来,分布式数据库的研究和应用有了很大的进展。

### 1) 多数据库系统

多数据库系统(Multi Basedata System, MBS)是为在多个数据库之间实现互操作,解决数据资源共享的一种技术途径。多数据库系统是由若干数据库组成的一个集合,其中每个数据库称为分数据库。分数据库系统可以是集中式的,或是分布式的。它们都受各自的、可能是不同的 DBMS(DataBase Management System, 数据库管理系统)管理。如果所有的分数据库系统都是相同的数据库管理系统,则称其为同构型的多数据库系统,否则称为异构型多数据库系统。多数据库系统不考虑全局模式,所有分数据库系统都是自治的,即数据定义自治,数据复制自治及在逻辑和物理级上的数据重构自治等。对多数据库系统来说,多数据库系统语言很重要,它应当提供数据库语言和允许分数据库实现互操作的全部功能,支持用户非过程的操作。Sybase 和 Oracle 都提供分布式处理的功能,支持分布多数据库系统。

Sybase 采用客户/服务器的体系结构,支持在网络环境下应用的分布计算模式。客户部分包括客户应用程序开发工具和客户接口。客户工具为应用软件开发的各个阶段提供支持,客户接口实现客户与服务器间的连接,以及开放客户的开放互连。服务器分为 SQL Server 和 Open Server 两部分。SQL Server 完成分布式 RDBMS(Relation DataBase Management System, 关系型数据库管理系统)的功能,Open Server 完成与其他非 SQL Server 管理的数据源和各种事务处理应用系统的互连。

Oracle 主要通过 SQL\*Net 和它的 RDBMS 实现分布式处理。它实现多点分布式查询,即一个查询可以涉及多个节点的数据库。Oracle 对多种平台、多种网络、多种操作系统的支持能力很强,特别在 Oracle 环境内,它提供的 4GL 工具具有比较强的分布查询能力。1992 年 Oracle 公司发布了 Oracle 7。从该产品起便采用协调服务器技术,提高分布检索功能和数据复制功能。

### 2) 联邦数据库系统

联邦数据库系统是多个协作数据库系统的集合,其中每个数据库系统都是自治的,也可能是异构的。组成的各个数据库系统可以在不同程度上集成。控制和协调操纵组成数据库系统的软件称为联邦数据库管理系统(Federated DataBase Management System, FDBMS)。联邦数据库系统的特点是分布、异构和自治。联邦数据库系统的体系结构一般采用以下 5 级模式:

- ✧ 局域模式;
- ✧ 组成模式;
- ✧ 联邦模式;