PowerBuilder 8.0
数据库应用系统开发教程

崔　巍　编著

清华大学出版社

需要全本请访问 www.ertongbook.com

(京) 新登字 158 号

内 容 简 介

本书以 PowerBuilder 为工具和环境，介绍了如何利用可视化技术、面向对象技术及组件技术等开发数据库应用系统，使读者了解当前计算机应用和计算机应用开发工具的最新发展，了解可视化、面向对象技术的开发方法和特点。

本书由三部分组成。第一部分是绪论(1 ～3 章)，主要内容包括数据库应用模式的发展和变化，数据库应用系统的开发方法和 PowerBuilder 简介。第二部分是 PowerBuilder 应用基础(4 ～9 章)，主要内容包括 PowerBuilder 的数据库操作，PowerScript 语言，用户界面设计，数据窗口技术，应用程序调试与发行等。第三部分是 PowerBuilder 的高级应用(10 ～14 章)，主要内容包括用户对象，动态数据窗口，一些高级控件的使用和 Web 开发技术等。

本书可以作为高等院校计算机专业和部分非计算机专业学习数据库开发及其工具的教材，可以作为学习 PowerBuilder 的培训教材，也可以供广大数据库应用系统开发人员阅读、参考。

版权所有，翻印必究。
本书封面贴有清华大学出版社激光防伪标签，无标签者不得销售。

书　　名：PowerBuilder 8.0 数据库应用系统开发教程
作　　者：崔　巍
出 版 者：清华大学出版社(北京清华大学学研大厦，邮编 100084)
　　http: //www.tup.tsinghua.edu.cn
责任编辑：章忆文
印 刷 者：清华大学印刷厂
发 行 者：新华书店总店北京发行所
开　　本：787× 1092 1/16　印张：21.5　字数：510 千字
版　　次：2002 年 1 月第 1 版　2002 年 1 月第 1 次印刷
书　　号：ISBN 7-302-05195-X/TP・3047
印　　数：0001 ～5000
定　　价：29．00 元

前　言

数据库技术始于20世纪60年代，迄今为止已经经历了30多年的发展。数据库的应用领域越来越广泛，数据库技术也将是所有信息技术和信息产业的基础。

国内的数据库应用广泛始于20世纪80年代，大部分中、小型应用都经历过Foxbase/Foxpro时代。但是随着计算机技术和应用需求的发展，数据库应用的模式已经由原来的单机/主机模式、网络文件服务器模式发展到现在的客户/服务器模式、分布式计算模式和浏览器/服务器模式等，相应的运行环境和开发工具都有了很大变化。

计算机软件开发是一项软件工程，早在20世纪60年代就提出了“软件危机”的问题，几十年来人们一直在为软件的开发速度、产品的可用性、可扩充性、可重用性、可维护性等努力着。20世纪70年代提出了结构化编程；后来又有了面向对象程序设计方法；近几年又大力推广即插即用的组件开发方法，这些技术具有互补性。20世纪90年代以来流行的一些可视化软件开发工具基本都支持以上各种方法，如PowerBuilder、Delphi、Visual Basic等。

作为一个学习计算机的大学生(计算机专业和部分非计算机专业学生)，在学习了程序设计课程之后，不能停留在“会编程”的水平上，还应该掌握一些开发工具，在走出校门前就具备开发应用系统的基本能力。

本书以PowerBuilder为工具和环境，介绍了如何利用可视化技术、面向对象技术及组件技术等开发应用系统，使读者了解当前计算机应用和计算机应用开发工具的最新发展，了解可视化、面向对象技术的开发方法和特点。

本书由三部分组成。第一部分是绪论(由1～3章组成)，第1章介绍了数据库应用模式的发展和变化；第2章概述了数据库应用系统的开发方法；第3章介绍了PowerBuilder的特点和特征。第二部分是PowerBuilder应用基础(由4～9章组成)，第4章介绍了PowerBuilder自带的数据库环境及其作用，以及在数据库画板中连接各种数据库和操作数据库的方法等；第5章介绍了PowerBuilder的编程语言PowerScript；第6章以一个简单的例子介绍了用PowerBuilder开发应用系统的方法，通过这一章读者可以对如何利用PowerBuilder开发应用系统建立起一个比较完整的概念；第7章介绍用户界面设计的内容，其中介绍了用于界面设计的各种控件的用途和使用方法、各种界面风格和菜单等；第8章介绍数据窗口技术，数据窗口是访问数据库的组件，它为开发数据库应用提供了非常便利的方法；第9章介绍PowerBuilder应用程序的调试方法和发行。第三部分是PowerBuilder的高级应用(由10～14章组成)，第10章介绍用户对象，它是可重用组件开发的基础；第11章介绍动态数据窗口应用技术；第12章介绍了一些PowerBuilder高级控件的使用方法，使用这些控件可以设计出更通用、更高级的应用程序；第13章介绍了一些常用的编程技术，如注册表的使用、Blob数据的处理、API函数的调用等；第14章介绍了PowerBuilder的Web开发技术。

本教材的教学目标是使同学们了解最新的数据库应用系统开发方法和工具，初步掌握面向对象、可视化、组件技术等代表目前先进开发思想的编程技术，为同学们今后参加信息系统和数据库应用项目的开发打下一个良好的基础。

另外，由于 PowerBuilder 8.0 的内容非常丰富，不可能在一本 50 万字左右的书中面面俱到，也不可能很详尽，如果读者通过本书的学习能够基本掌握现代数据库应用的开发方法和开发工具，掌握 PowerBuilder 的开发思路和开发技术，能够为以后的数据库应用开发工作奠定一个良好的基础，那么笔者就已经甚感欣慰了。

由于时间仓促和作者水平有限，书中疏漏之处在所难免，欢迎广大读者批评指正。

作　者

目　　录

第一部分　绪　　论

第 1 章　数据库应用模式的发展 …… 2
1.1　主机应用模式 …… 2
1.2　文件服务器应用模式 …… 3
1.3　客户/服务器应用模式 …… 4
1.4　文件服务器与客户/服务器的数据库操作 …… 5
1.5　分布式计算应用模式 …… 6
1.6　Web 网络应用模式 …… 8
习题 …… 10

第 2 章　数据库应用系统开发方法概述 …… 11
2.1　结构化生命周期法 …… 11
2.2　快速原型法 …… 13
2.3　面向对象方法 …… 14
2.3.1　面向对象的基本概念 …… 14
2.3.2　面向对象的分析与设计 …… 16
2.3.3　面向对象技术在 PowerBuilder 中的实现 …… 17
2.4　PowerBuilder 的开发方法 …… 19
习题 …… 20

第 3 章　PowerBuilder 简介 …… 21
3.1　PowerBuilder 的特点 …… 21
3.2　owerBuilder 7.0 曾有哪些改进 …… 24
3.2.1　用户界面和开发方法的变化 …… 24
3.2.2　新的组件开发和提交手段 …… 25
3.2.3　新的数据库连接 …… 26
3.2.4　其他新特性 …… 27
3.3　owerBuilder 8.0 的新特性 …… 28
3.3.1　卓越的应用开发效率 …… 28
3.3.2　全新的 Web 特性和强大的 Web 开发功能 …… 30
3.3.3　与 EAServer 的紧密集成 …… 31
3.3.4　数据库连接方面新的变化 …… 32

需要全本请访问 www.ertongbook.com

3.3.5 其他改进 …… 33
3.4 owerBuilder 对象概述 …… 35
3.4.1 对象、属性与事件 …… 35
3.4.2 PowerBuilder 库(Library) 与文件类型 …… 35
3.4.3 应用对象(Application) …… 36
3.4.4 窗口对象(Window) …… 37
3.4.5 数据窗口对象(DataWindow) …… 37
3.4.6 菜单(Menu) …… 39
3.4.7 全局函数(Global functions) …… 39
3.4.8 查询(Query) …… 39
3.4.9 结构(Structure) …… 39
3.4.10 用户对象(User Object) …… 40
3.4.11 工程对象(Project) …… 40
3.5 owerBuilder 8.0 的开发环境 …… 40
3.5.1 工作空间和目标 …… 40
3.5.2 PowerBuilder 窗口 …… 41
3.5.3 系统树、剪贴和输出窗口 …… 41
3.5.4 画板(Painter) …… 43
3.5.5 PowerBar、PainterBar 和 StyleBar …… 43
3.5.6 定制工具栏 …… 47
习题 …… 48

第二部分 PowerBuilder 应用基础

第 4 章 数据库操作 …… 50
4.1 连接数据库 …… 50
4.1.1 定义数据源 …… 50
4.1.2 配置数据库连接参数 …… 52
4.2 操作数据库 …… 54
4.2.1 浏览数据 …… 55
4.2.2 插入、删除和更新操作 …… 56
4.2.3 浏览和修改表结构 …… 58
4.2.4 删除表 …… 60
4.3 创建一个新的数据库 …… 60
4.3.1 创建数据库 …… 60
4.3.2 创建表 …… 64
4.3.3 建立关键字和索引 …… 66
4.4 断开与数据库的连接 …… 68

习题 …… 69

第 5 章 PowerScript 语言简介 …… 70
5.1 基本概念 …… 70
5.1.1 注解 …… 70
5.1.2 标识符 …… 71
5.1.3 特殊的 ASCII 字符 …… 71
5.1.4 保留字 …… 72
5.1.5 代词 …… 73
5.1.6 续行 …… 75
5.1.7 语句分隔符 …… 76
5.1.8 空值 …… 76
5.2 数据类型 …… 77
5.2.1 标准数据类型 …… 77
5.2.2 任意(Any) 数据类型 …… 79
5.2.3 系统对象数据类型 …… 80
5.2.4 枚举数据类型 …… 81
5.3 说明 …… 82
5.3.1 说明变量 …… 82
5.3.2 说明常量 …… 85
5.3.3 数组的说明和使用 …… 85
5.4 运算符和表达式 …… 87
5.5 结构和对象 …… 88
5.5.1 结构 …… 88
5.5.2 对象 89
5.6 owerScript 语句 …… 90
5.6.1 赋值语句 …… 90
5.6.2 条件语句 …… 91
5.6.3 多重分支语句 …… 92
5.6.4 循环语句 …… 94
5.6.5 GOTO 语句 …… 96
5.6.6 CALL 语句 …… 96
5.6.7 CREATE 和 DESTROY 语句 …… 97
5.6.8 HALT 语句 …… 98
5.6.9 RETURN 语句 99
5.7 调用函数和事件 …… 99
5.7.1 函数和事件的一些概念 …… 99
5.7.2 调用函数和事件 …… 100

5.7.3 调用祖先对象中的函数和事件 …… 102
5.7.4 关于系统函数 …… 103
5.8 嵌入 SQL 与数据库操作 …… 104
5.8.1 连接数据库的语句 …… 104
5.8.2 SELECT 查询语句 …… 106
5.8.3 操作语句 …… 107
5.8.4 事务提交与撤消 …… 109
5.8.5 使用游标的语句 …… 110
5.8.6 利用存储过程进行查询的语句 …… 112
5.8.7 查询和更新 Blob 类型字段的语句 …… 114
5.8.8 动态 SQL 语句 …… 116
习题 …… 125

第 6 章 第一个 PowerBuilder 应用程序 …… 126
6.1 从查询单个表开始 …… 126
6.1.1 建立工作空间 …… 126
6.1.2 建立目标和应用对象 …… 127
6.1.3 建立窗口对象 …… 129
6.1.4 建立数据窗口对象 …… 130
6.1.5 修改数据窗口对象 …… 132
6.1.6 增加控件 …… 136
6.1.7 写事件驱动程序 137
6.2 增加排序功能 …… 139
6.2.1 添加控件 …… 139
6.2.2 有关排序的函数 …… 140
6.2.3 写事件驱动程序 141
6.3 关联与条件查询 …… 142
6.3.1 带参数的数据窗口对象 …… 142
6.3.2 增加一个数据窗口控件 …… 145
6.3.3 写事件驱动程序 145
6.4 定制 PowerBuilder 环境 …… 146
6.4.1 钉住窗口标题条 …… 147
6.4.2 移动和复位窗口 …… 147
6.4.3 保存窗口布局模式 …… 148
6.4.4 设置启动选项 148
习题 149

第 7 章 用户界面设计 …… 150
7.1 窗口与控件 …… 150
7.1.1 窗口对象 …… 150
7.1.2 窗口上的控件 …… 153
7.1.3 窗口设计中的常用技术 166
7.2 常用对话框 …… 170
7.2.1 消息对话框 …… 171
7.2.2 Open 对话框 …… 172
7.2.3 Save 对话框 174
7.3 菜单 …… 175
7.3.1 建立菜单 …… 175
7.3.2 为窗口指定菜单 179
7.4 DI 界面 …… 179
7.4.1 MDI 窗口的构成 …… 179
7.4.2 建立 MDI 窗口 …… 180
7.4.3 MDI 窗口的工具栏 …… 181
7.4.4 MDI 窗口的 MicroHelp 183
7.5 多窗口实例 …… 184
7.5.1 窗口对象与实例 …… 184
7.5.2 多窗口实例的使用 …… 185
7.5.3 窗口数组 …… 185
习题 …… 187

第 8 章 数据窗口 …… 188
8.1 数据窗口对象 …… 188
8.1.1 数据窗口基础 …… 188
8.1.2 数据窗口的显示风格 …… 189
8.1.3 数据窗口的数据源 …… 193
8.1.4 使用数据窗口的基本步骤 196
8.2 数据窗口画板及其操作 …… 196
8.2.1 数据窗口画板环境 …… 196
8.2.2 Design 窗口 …… 197
8.2.3 Preview 窗口与数据库操作 198
8.3 设计数据窗口对象 …… 199
8.3.1 设置数据窗口对象的属性 …… 199
8.3.2 向数据窗口对象添加控件 …… 202
8.3.3 添加计算列或计算域 205
8.4 数据窗口控件 …… 207

8.4.1 事务对象与数据库连接 …… 208
8.4.2 为数据窗口指定事务对象 …… 211
8.4.3 数据窗口控件的操作 212
8.5 数据存储对象 …… 218
习题 218

第 9 章 程序的调试与发布 …… 220
9.1 调试应用程序 …… 220
9.1.1 Debug 画板 …… 220
9.1.2 在程序中设置断点 …… 222
9.1.3 在调试模式下执行程序 …… 223
9.1.4 观察变量的值 …… 224
9.1.5 设置条件断点 224
9.2 编译与发行应用程序 …… 225
9.2.1 生成可执行程序 …… 226
9.2.2 发布应用程序 228
习题 228

第三部分 PowerBuilder 高级应用

第 10 章 可重用部件的设计 …… 230
10.1 用户对象概述 …… 230
10.1.1 类用户对象 …… 230
10.1.2 可视用户对象 231
10.2 建立用户对象 …… 232
10.2.1 用户对象画板环境 …… 233
10.2.2 建立定制类用户对象 …… 234
10.2.3 建立标准类用户对象 …… 235
10.2.4 建立定制可视用户对象 …… 235
10.2.5 建立标准可视用户对象 235
10.3 用户事件 …… 236
10.3.1 定义用户事件 …… 236
10.3.2 为用户对象的事件编写程序 238
习题 242

第 11 章 动态数据窗口 …… 243
11.1 动态指定数据窗口对象 …… 243
11.2 动态建立数据窗口对象 …… 244

11.3 动态修改数据窗口对象 …… 247
11.3.1 添加对象到数据窗口 …… 248
11.3.2 删除数据窗口中的对象 …… 249
11.3.3 修改数据窗口对象的属性 249
11.4 QBE 数据窗口 …… 250
习题 252

第 12 章 几个高级控件的应用 …… 253
12.1 Tab 标签控件 …… 253
12.1.1 术语及实例说明 …… 253
12.1.2 建立 Tab 控件和选项卡 …… 254
12.1.3 管理 Tab 控件及其选项卡 …… 255
12.1.4 写 Tab 控件的程序 256
12.2 istView 控件 …… 258
12.2.1 在画板中设计列表窗口 …… 259
12.2.2 ListViewItem 对象 …… 260
12.2.3 ListView 控件的函数 …… 260
12.2.4 在程序中构造列表窗口中的项目 …… 262
12.2.5 ListView 控件的详细资料方式 264
12.3 reeView 控件 …… 266
12.3.1 TreeView 控件概述 …… 267
12.3.2 TreeView 控件应用举例 …… 267
12.3.3 TreeView 控件应用技术 269
12.4 raph 控件及应用 …… 277
12.4.1 Graph 控件简介 …… 277
12.4.2 Graph 控件应用举例 …… 278
12.4.3 动态改变图形的类型 284
习题 286

第 13 章 几种常用编程技术 …… 287
13.1 使用初始化文件和 Windows 注册表 …… 287
13.1.1 使用初始化文件 …… 287
13.1.2 使用 Windows 注册表 289
13.2 处理 Blob 数据 …… 291
13.2.1 处理 Blob 数据的 SQL 语句 …… 291
13.2.2 利用 Blob 数据完成对图片或大文本的处理 292
13.3 动态链接库与外部函数的调用 …… 293
13.3.1 说明外部函数 …… 294

需要全本请访问 www.ertongbook.com

13.3.2 外部函数调用举例 295
13.4 数据转移技术 …… 298
13.4.1 数据管道画板 …… 298
13.4.2 数据管道程序设计 299
13.5 ichText 应用 …… 305
13.5.1 RichTextEdit 控件的使用 …… 305
13.5.2 RichTextEdit 控件中的输入域 …… 308
13.5.3 预览和打印 …… 309
13.5.4 使用数据库中的数据 …… 311
13.5.5 RichText 数据窗口 312
习题 313

第 14 章 一个简单的 Web 应用 …… 314
14.1 建立 PowerDynamo Web 站点 …… 314
14.2 建立和修改基本的 Web 主页 …… 315
14.2.1 建立 4GL 的 Web 主页 …… 316
14.2.2 修改主页的外观 …… 318
14.2.3 添加一个图片 …… 319
14.2.4 为图片指定一个绝对位置 320
14.3 添加主页导航 …… 322
14.3.1 为 Welcome.htm 主页添加一个超链接 …… 322
14.3.2 为 Product.htm 主页添加一个按钮 323
14.4 完成注册主页的功能 …… 324
14.4.1 添加会话变量 …… 324
14.4.2 添加单行文本控件 …… 325
14.4.3 添加口令验证 …… 326
14.4.4 添加服务器重定向 328
14.5 指定一个起始主页 …… 329
14.6 编译和运行 Web 应用程序 330

第一部分

绪　　论

第一部分由以下 3 章组成:

数据库应用模式的发展

数据库应用系统开发方法概述

PowerBuilder 简介

其中第 1 章介绍了数据库应用模式由主机应用模式、文件服务器应用模式、客户/服务器应用模式到分布式应用模式和 Web 网络应用模式的发展和变化; 第 2 章概述了数据库应用系统的开发方法, 如结构化生命周期方法、快速原型方法、面向对象方法以及利用 PowerBuilder 进行系统开发的方法等; 第 3 章概要介绍了 PowerBuilder 的特点和特征, 包括 PowerBuilder 的一般特性、PowerBuilder 8.0 的新的变化和新的功能、PowerBuilder 的各种对象及 PowerBuilder 的开发环境等。

第 1 章　数据库应用模式的发展

PowerBuilder 是一个支持面向对象技术、支持客户/ 服务器机制、支持分布式组件开发、支持 Web 应用的数据库应用系统开发工具，所以我们有必要首先了解一下数据库应用的模式及其演变过程。

计算机的应用结构经历了集中式结构、文件服务器的网络结构到现在的客户/ 服务器网络结构、分布式客户/ 服务器网络结构和 Web 网络结构的发展阶段。我们通过概述这几种结构的特点，来充分理解什么是客户/ 服务器网络结构、分布式客户/ 服务器网络结构和 Web 网络结构，以及为什么使用这些结构等。

1.1　主机应用模式

在 20 世纪 60—70 年代，真正需要使用计算机的企业都会考虑使用大型机，它代表一种“集中式”的系统结构。它由两个关键硬件组成：主机和客户终端，如图 1-1 所示，这里的主机相当于现在的服务器。

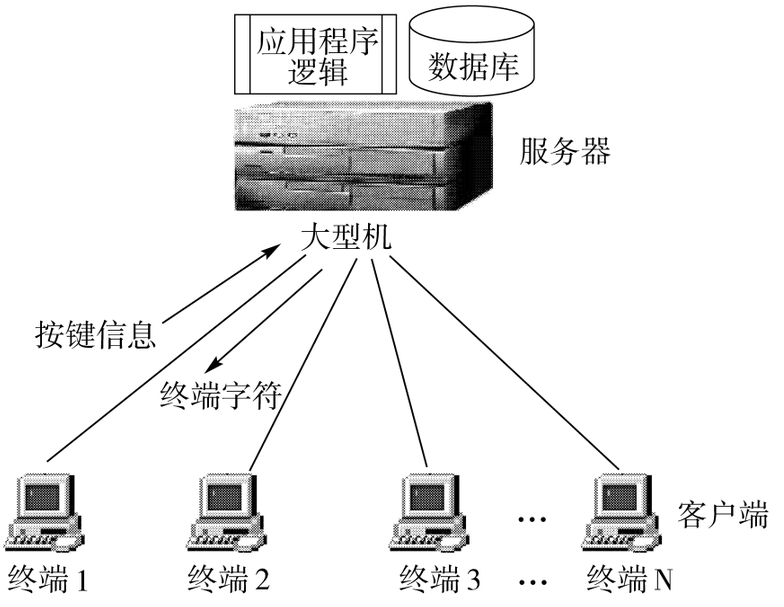


图 1-1　集中式主机结构

在集中式结构中，客户终端和主机之间传递数据的方式非常简单，一是用户从客户终端键盘键入信息到主机，二是由主机返回到终端上的字符。这个时期的计算机的所有资源(数据) 都在主机上，所有处理(程序) 也在主机上完成。这种结构的优点是可以实现集中管理，安全性很好。这种计算机的费用非常昂贵，并且应用程序和数据库都存放在主机中，没有办法真正划分应用程序的逻辑。

1.2　文件服务器应用模式

到 20 世纪 80 年代，个人计算机进入了商用舞台，同时计算机应用的范围和领域也日趋广泛。这对那些没有能力实现大型机方案的企业来说，个人计算机无疑就有了用武之地。在个人计算机进入商用领域不久，局域网也问世了，同时也诞生了文件服务器技术，图 1-2 示意了文件服务器结构。

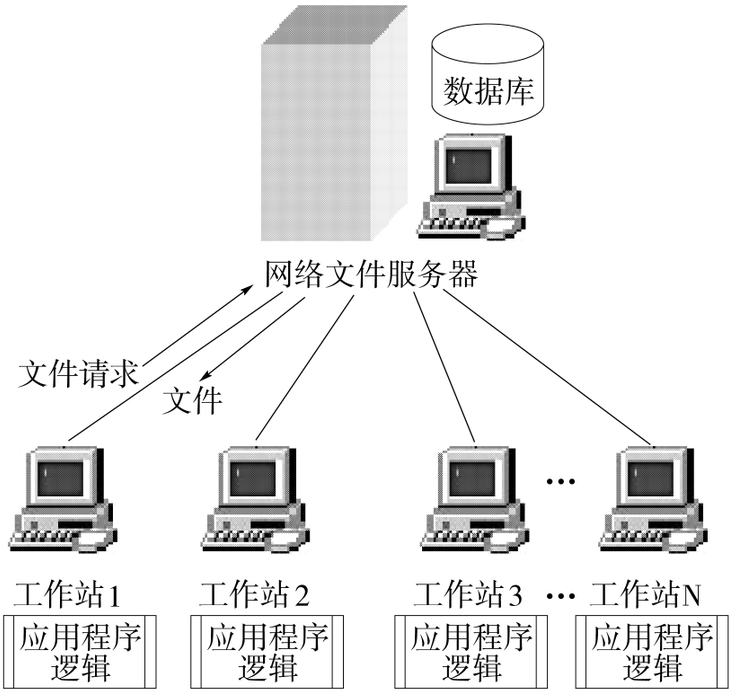


图 1-2　文件服务器结构

从图 1-2 可以看出，在文件服务器系统结构中，应用程序是在客户工作站上运行的，而不是在服务器上运行的，文件服务器只提供了资源(数据)的集中管理和访问途径。这种结构的特点是将共享数据资源集中管理，而将应用程序分散安排在各个客户工作站上。文件服务器结构的优点在于实现的费用比较低廉，而且配置非常灵活，在一个局域网中可以方便地增减客户端工作站。但是，文件服务器结构的缺点也非常明显，由于文件服务器只提供文件服务，所有的应用处理都要在客户端完成，这就意味着客户端的个人计算机必须要有足够的能力，以便执行需要的任何程序，或能完成任何必要的任务。这可能经常需要客户端的计算机升级，否则改进应用程序的功能、提高应用程序的性能等都会成为一句空话。

特别要提出的是，虽然应用程序可以存放在网络文件服务器的硬盘上，但它每次都要传送到客户端的个人计算机的内存中执行。另外，所有的处理都是在客户端完成的，网络上就要经常传送大量无用的数据。

1.3 客户/服务器应用模式

文件服务器结构的费用虽然低廉，但是和大型机的“集中式”相比，它缺乏足够的计算和处理能力。为了解决费用和性能的矛盾，客户/服务器结构就应运而生了。这种结构允许应用程序分别在客户工作站和服务器(注意：不再是文件服务器)上执行，可以合理划分应用逻辑，充分发挥客户工作站和服务器两方面的性能。图1-3示意了客户/服务器的结构。

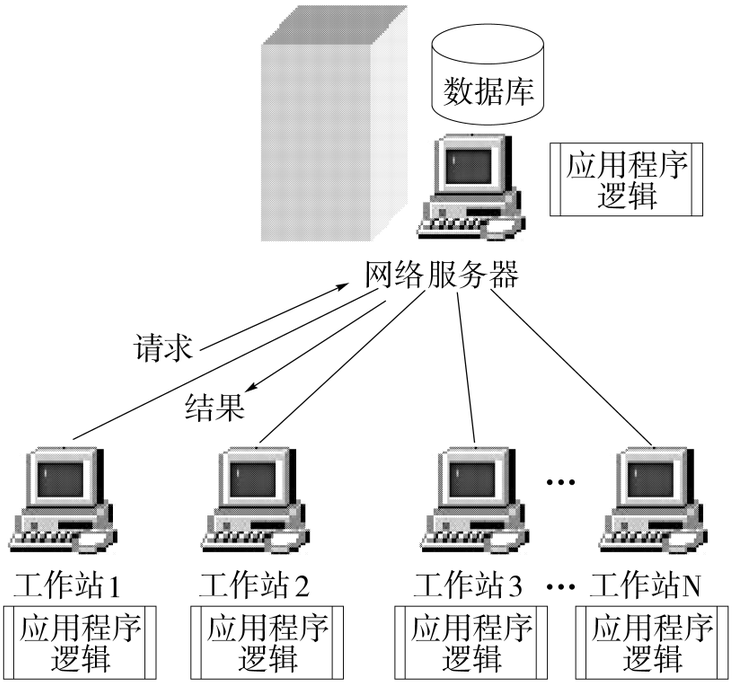


图1-3 客户/服务器结构

在客户/服务器结构中，应用程序或应用逻辑可以根据需要划分在服务器和客户工作站中。这样，为了完成一个特定的任务，客户工作站上的程序和服务器上的程序可以协同工作。从图1-3可以看出客户/服务器结构和文件服务器结构的区别：客户/服务器结构的客户工作站向服务器发送的是处理请求，而不是文件请求；服务器返回的是处理的结果，而不是整个文件。

当今，客户/服务器最流行的领域就是数据库应用领域。比较著名的数据库厂商都提供了支持客户/服务器结构的数据库管理系统，如Microsoft的SQL Server、Sybase的Adaptive Server和Oracle等。

从以上我们可以看出，大型机集中式结构的所有程序都在主机内执行，而文件服务器局域网结构的所有程序都在客户端执行，这两种结构都不能提供真正的可伸缩应用系统框架。而客户/服务器结构则可以将应用逻辑分布在客户工作站和服务器之间，可以提供更快、更有效的应用程序性能。

在客户/服务器结构中，我们常把客户端称做前台或前端客户，把服务器称做后台

或后端服务器。

1.4　文件服务器与客户 / 服务器的数据库操作

Microsoft 的 Access 和 Foxpro 都是非常流行的数据库管理系统，它们的数据库都可以建立在服务器上，也允许多用户访问，但它们绝对不是客户 / 服务器方案，它们是文件服务器方案。为此，我们需要搞清楚文件服务器方案和客户 / 服务器方案的区别，如果对它们之间的差异没有清醒的和正确的认识，即便有支持客户 / 服务器方案的环境，也不可能充分利用客户 / 服务器结构为我们提供强大的功能。图 1-4 和图 1-5 示意了这两种结构之间的差异。我们假设在数据库上都有一个含有 30000 条记录的历届学生的数据库表格 student(并且进一步假设是 Foxpro 的 DBF 格式文件)，并且要求查询学号为 0121116 的学生的信息。

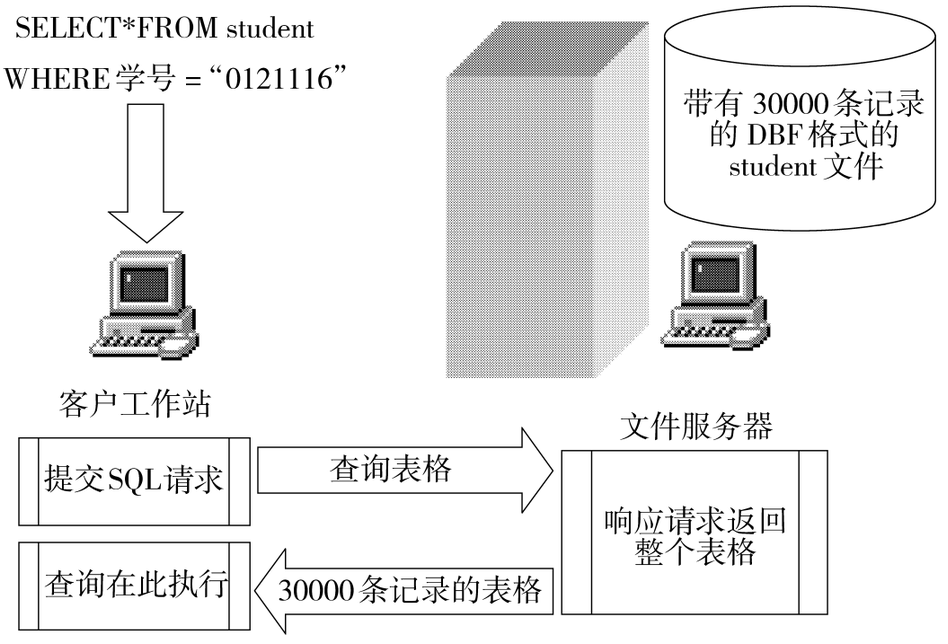


图 1-4　访问文件服务器的数据库

图 1-4 是文件服务器结构，查询根本不会传送给服务器，查询是在客户端完成的。客户端的系统根据访问数据库的要求，它“意识”到自己需要一张数据表，以便进行查询和得出结果。因此，在正式执行 SELECT 语句之前，查询逻辑会请求通过网络将 30000 行的表格传送到本地的客户工作站上，然后在客户端进行处理，完成查询。如果需要查找的记录不存在，则网络传输实际是在做无用功。由此看来，文件服务器只是负责文件的集中管理，并根据客户端的请求向客户端发送文件，除此之外文件服务器不会执行其他任何程序逻辑。显然，文件服务器的处理方式会增加网络线路的传输负荷，降低传输的效率和响应时间，很容易造成网络阻塞。

图 1-5 是客户 / 服务器结构，实际的 SQL 查询语句将在服务器中执行，服务器发送给客户端工作站的只是查询的结果。也就是说，客户端的应用程序向数据库服务器(即运行在服务器端的数据库管理系统) 发出的请求是“传送数据库表 student 中学号是

需要全本请访问 www.ertongbook.com