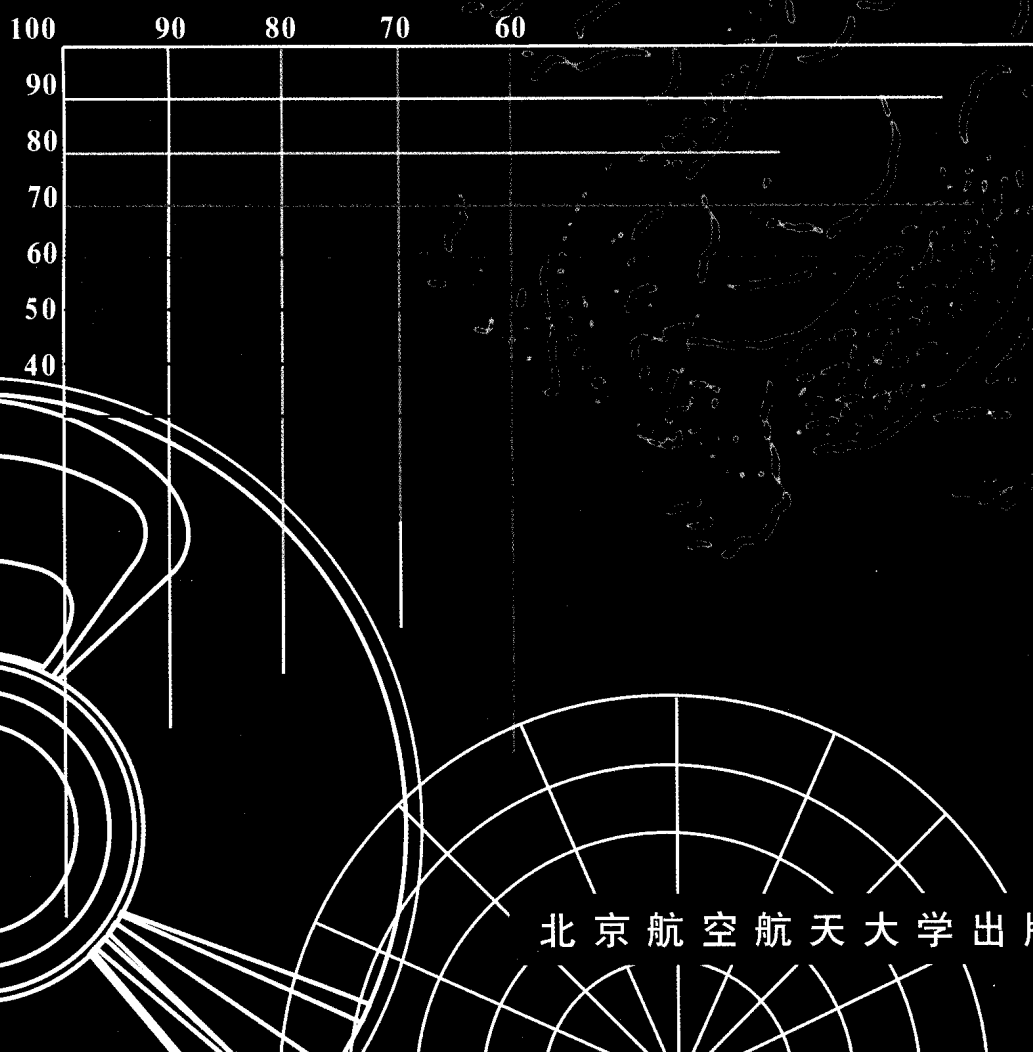


■ 洪 星 罗 辉 编著

中文
版

Visual FoxPro 5.0



北京航空航天大学出版社

内 容 提 要

中文版 Visual FoxPro 5.0 是微软公司于 1997 年推出的优秀的数据库开发平台。本书作者将根据自己的数据库开发经验,以丰富、详细而实用的实例程序、各种特殊的设计技巧,详尽地介绍中文版 Visual FoxPro 5.0 的可视编程环境和面向对象的开发技术。

本书不同于其它中译本,它完全以适合我国读者口味的方式组织整个章节,可读性强,示例新颖实用,具有一定代表性,而且指明原系统存在的几个“bug”。本书不论是对初学者,还是对数据库开发专家,不论是对 FoxPro 2.5 用户,还是对 Visual FoxPro 3.0 用户,都具有很强的参考价值。

图书在版编目(CIP)数据

中文版 Visual FoxPro 5.0 应用开发技术/洪星等编著. —
北京:北京航空航天大学出版社,1998.8
ISBN 7-81012-805-1

I. 中… II. 洪… III. 关系数据库-数据库管理系统, FoxPro5.0-程序设计 IV. TP311.13

中国版本图书馆 CIP 数据核字 (98) 第 14361 号

中文版 Visual FoxPro 5.0 应用开发技术

洪 星 罗 辉 编著

责任编辑 赵延永

责任校对 陈 坤

*

北京航空航天大学出版社出版发行

(北京市学院路 37 号 100083), 发行部电话 010-62015720)

http://www.buaapress.cn.net

E-mail: pressell@publica.bj.cninfo.net

北京朝阳科普印刷厂印装 各地书店经销

*

开本: 787×1092 1/16 印张: 25.5 字数: 652.8 千字

1998 年 7 月第 1 版 1998 年 7 月第 1 次印刷 印数: 4000 册

ISBN 7-81012-805-1/TP·294 定价: 34.00 元

前 言

如果想迅速提高在中文版 Visual FoxPro 5.0 上的应用和开发水平,你应该仔细研读手头的这本书。

本书作者多年来从事数据库应用系统的开发,熟悉 Informix, SQL Server, Sybase 等大型数据库系统,对 Visual Basic, Delphi 等面向对象前端数据库开发工具有丰富的开发经历,对 FoxPro 系列产品的功能更是了如指掌。本书内容的组织和编排,就是根据作者多年来摸索 Visual FoxPro 数据库开发技术的实际过程和经验逐步完成的。

中文版 Visual FoxPro 5.0 是微软公司于 1997 年推出的 Visual Studio 系列新产品之一。它在中文版 Visual FoxPro 3.0 的基础上增加了许多新的功能和特性,但是其基础框架仍与 Visual FoxPro 3.0 保持一致,因此许多功能、性能及使用上与 Visual FoxPro 3.0 是类似的。可以说,中文版 Visual FoxPro 5.0 是中文版 Visual FoxPro 3.0 的增强版。

本书是针对中文版 Visual FoxPro 5.0 的诸多开发环境、开发工具,系统而详尽地展开介绍的。它根据作者在中文版 Visual FoxPro 5.0 系统中的应用和开发经历,用 14 章的篇幅介绍了中文版 Visual FoxPro 5.0 的应用开发技术。透过大量的实例,读者可以了解到系统许多秘而不宣的技巧、特殊的编程经验、简捷的操作方法。本书不同于一般的中译本技术书籍,它通篇极赋技巧性、实用性和可读性。

希望精通中文版 Visual FoxPro 5.0 系统的操作和程序设计的用户,本书将为你提供许多解决问题的技巧和经验;凡对 FoxPro 系列产品有基本印象的用户,通过它可以迅速加深对系统的理解,提高对系统的操作和编程能力,获得许多应用技巧和方法;对于非 Visual FoxPro 5.0 的用户,本书也将带给你诸多有益的启迪。

本书编写过程中,得到了湖南省计生委和湖南省工商银行系统领导的殷切关怀和支持,并特别得到了张芙蓉副主任、尹晖处长、信息中心周松茂副主任的直接关心和指导,也得到了英宝、刘芝文等同志的默契配合,在此一并致谢。

作 者

1998 年 2 月于湖南长沙



目 录

第一章 Visual FoxPro 5.0 概述	(1)
1.1 完善的数据库引擎.....	(1)
1.1.1 引入了数据辞典.....	(1)
1.1.2 增强的数据库管理.....	(1)
1.1.3 加强数据完整性验证机制.....	(2)
1.1.4 扩充的数据类型.....	(2)
1.1.5 数据环境、表和关联成为对象	(2)
1.1.6 使用游离视图增强连接能力.....	(2)
1.2 引进了先进的程序设计技术.....	(2)
1.2.1 面向对象编程.....	(2)
1.2.2 可视化编程.....	(3)
1.3 众多强大的设计工具.....	(4)
1.3.1 增强的表单设计器和代码编辑器.....	(4)
1.3.2 ActiveX 控件提供了强大的外援	(4)
1.3.3 数据库设计器和表设计器.....	(5)
1.3.4 增强的查询及视图设计.....	(5)
1.3.5 类设计器和类浏览.....	(5)
1.3.6 加强了的向导程序.....	(6)
1.3.7 已改善的调试工具.....	(6)
1.3.8 应用程序组件的实例.....	(7)
第二章 进入 Visual FoxPro 5.0	(8)
2.1 Visual FoxPro 5.0 的安装	(8)
2.1.1 安装 Visual FoxPro5.0 的环境要求	(8)
2.1.2 安装 Visual FoxPro 5.0	(8)
2.1.3 更新 Visual FoxPro 3.0 的配置文件	(9)
2.2 Visual FoxPro 5.0 集成开发环境	(9)
2.2.1 启动 Visual FoxPro 5.0	(9)
2.2.2 退出 Visual FoxPro 5.0	(11)
2.2.3 Visual FoxPro 5.0 集成开发环境	(11)
2.3 Visual FoxPro 5.0 参数设置	(16)
2.3.1 多个应用程序主窗口属性的协调.....	(16)
2.3.2 临时文件及其存放.....	(17)
2.3.3 参数配置文件 CONFIG.FPW	(17)
2.4 字体的使用	(19)
2.4.1 字 体.....	(20)

2.4.2	FoxFont 字体	(20)
2.4.3	等宽字体和比例字体	(20)
2.4.4	确定可使用的字体	(21)
2.4.5	磅和字号的换算	(21)
2.5	本章小结	(22)
第三章	系统菜单和工具栏	(23)
3.1	控制菜单	(23)
3.2	“文件”菜单	(23)
3.3	“编辑”菜单	(27)
3.4	“格式”菜单	(30)
3.5	“工具”菜单	(33)
3.5.1	调试器	(33)
3.6	“程序”菜单	(38)
3.7	“窗口”菜单	(39)
3.8	“帮助”菜单	(40)
3.9	“显示”菜单及工具栏	(40)
3.10	本章小结	(46)
第四章	数据库和表设计	(47)
4.1	有关数据库、表的基本概念	(47)
4.1.1	基本概念	(47)
4.1.2	字段的类型	(49)
4.1.3	索引标记类型	(50)
4.1.4	数据库的完整性规则	(50)
4.1.5	存储过程	(51)
4.2	数据库设计器和表设计器	(51)
4.2.1	数据库设计器	(51)
4.2.2	“数据库”菜单	(51)
4.2.3	数据库设计工具栏	(53)
4.3	用表设计器添加一个新表	(54)
4.3.1	字段页	(54)
4.3.2	字段级完整性验证规则	(55)
4.3.3	表 页	(57)
4.3.4	记录级完整性验证规则	(58)
4.3.5	索引页	(58)
4.3.6	数据表间的关联	(59)
4.3.7	关联表级参照完整性验证规则	(61)
4.3.8	建立一个自由表添加到数据库中	(72)

4.4	表记录数据的操作	(72)
4.4.1	“表”菜单	(72)
4.4.2	记录的追加及“虚记录”	(75)
4.4.3	Browse 窗口操作	(75)
4.5	用命令操作数据库和表	(76)
4.5.1	数据库和表的建立	(76)
4.5.2	数据库和表的打开与关闭	(78)
4.5.3	相关表和自由表的相互转换	(81)
4.5.4	显示当前数据库及其表的信息	(81)
4.5.5	指定当前数据库	(81)
4.6	存储过程的使用	(82)
4.6.1	存储过程的创建、修改和删除	(82)
4.6.2	使用远程服务器的存储过程	(84)
4.6.3	调用服务器存储过程返回多个结果集	(84)
4.6.4	服务器处理结果集和错误	(84)
4.7	本章小结	(85)
第五章	表单设计器	(86)
5.1	面向对象编程	(86)
5.1.1	对象和面向对象编程	(86)
5.1.2	屏幕界面的面向对象编程步骤	(86)
5.2	表单设计器环境	(87)
5.2.1	启动表单设计器	(87)
5.2.2	表单设计窗口	(88)
5.2.3	表 单	(88)
5.2.4	“表单”菜单和扩充的“显示”菜单	(89)
5.2.5	表单控件	(90)
5.2.6	表单工具栏	(91)
5.2.7	属性窗口	(91)
5.2.8	代码编辑器	(92)
5.2.9	数据环境窗口	(93)
5.3	屏幕控件	(95)
5.3.1	表单的基本控件分类	(95)
5.3.2	控件的使用	(96)
5.4	对 象	(98)
5.4.1	对象属性	(98)
5.4.2	对象事件	(98)
5.4.3	对象方法	(100)
5.4.4	设计对象的事件和方法处理代码的规则	(101)

5.4.5	ThisForm 和 This 对象	(101)
5.5	设计第一个实例	(102)
5.5.1	设计用户界面	(102)
5.5.2	设置属性	(103)
5.5.3	在事件过程中添加代码	(104)
5.6	实例程序的保存和运行	(105)
5.6.1	保存实例程序	(105)
5.6.2	运行实例程序	(106)
5.6.3	事件的响应	(106)
5.7	本章小结	(107)
第六章	基本屏幕控件的使用	(108)
6.1	表 单	(108)
6.1.1	表单具有的属性	(109)
6.1.2	表单响应的事件	(112)
6.1.3	表单支持的方法	(113)
6.1.4	表单的应用	(114)
6.1.5	特殊表单——主窗口的控制	(117)
6.2	直线、形状、标签和图像控件	(118)
6.2.1	直 线	(119)
6.2.2	形 状	(122)
6.2.3	标 签	(123)
6.2.4	图 像	(124)
6.3	文本框、编辑框和旋转器	(125)
6.3.1	文本框	(125)
6.3.2	编辑框	(128)
6.3.3	旋转器	(131)
6.4	命令按钮和按钮组	(133)
6.4.1	命令按钮	(133)
6.4.2	按钮组	(135)
6.5	检查框和选项框	(138)
6.5.1	检查框	(138)
6.5.2	选项框	(140)
6.6	列表框和组合列表框	(142)
6.6.1	列表框	(142)
6.6.2	组合列表框	(150)
6.6.3	列表框和组合列表框选项数据来源	(153)
6.7	生成器工具	(153)
6.8	本章小结	(154)

第七章 页框架、网格和定时器	(155)
7.1 页框架	(155)
7.1.1 页框架控件和页控件	(155)
7.1.2 页框架中页的引用	(155)
7.1.3 页框架和页的属性、事件和方法	(156)
7.1.4 “工具/选项”菜单项的使用	(157)
7.1.5 在页中放置公共控件和私有控件	(168)
7.1.6 页突的控制	(168)
7.1.7 页框架中页数的动态调整	(168)
7.2 网 格	(173)
7.2.1 网格、列、栏头	(173)
7.2.2 网格列和栏头的属性、事件和方法	(173)
7.2.3 对网格中每一个元素进行控制	(175)
7.3 定时器	(184)
7.3.1 定时器	(184)
7.3.2 定时器的属性、事件和方法	(184)
7.3.3 数字计时器	(184)
7.3.4 计时秒表	(185)
7.3.5 定时激活一个任务	(187)
7.4 本章小结	(188)
第八章 ActiveX 控件及其使用	(189)
8.1 OLE 容器控件和 OLE 绑定控件	(189)
8.1.1 OLE 容器控件	(189)
8.1.2 OLE 绑定型控件	(190)
8.1.3 通用型字段使用 OLE 对象	(190)
8.2 ActiveX 控件	(191)
8.3 影视播放器功能简介	(192)
8.4 影视播放器表单设计	(193)
8.4.1 建立表单及其属性	(193)
8.4.2 设置表单的事件处理代码	(193)
8.4.3 创建新方法程序	(195)
8.4.4 设置播放区及控制面板的控件和属性	(196)
8.4.5 添加 ActiveX 控件	(198)
8.4.6 设置控制面板中控件的事件	(199)
8.4.7 影视播放器的运行	(202)
8.5 外部 Windows 32 位函数的使用	(202)
8.6 本章小结	(203)

第九章 程序设计语言的使用	(204)
9.1 MessageBox() 函数	(204)
9.1.1 对话框的显示信息	(204)
9.1.2 对话框类型	(204)
9.1.3 对话框标题条文本	(205)
9.1.4 对话框返回值信息	(206)
9.2 空值(NULL)及其应用	(207)
9.2.1 何谓空值	(207)
9.2.2 ISNULL()、EMPTY() 和 ISBLANK() 函数	(207)
9.2.3 在数据表中的空值支持和置空白操作	(209)
9.2.4 空值的应用	(210)
9.3 过程和函数	(211)
9.3.1 过程或函数的代码形式	(211)
9.3.2 全局变量、局部变量和本地变量	(212)
9.3.3 过程或函数的调用	(214)
9.3.4 过程和函数的参数	(214)
9.4 位操作	(218)
9.4.1 按位与(AND)操作	(219)
9.4.2 按位非(NOT)操作	(219)
9.4.3 按位或(OR)操作	(219)
9.4.4 按位异或(XOR)操作	(220)
9.4.5 按位置零和置 1 操作	(220)
9.4.6 按位左移和右移操作	(221)
9.4.7 位测试操作 BITTEST() 函数	(221)
9.4.8 位操作在加密中的作用	(222)
9.4.9 扩充位操作函数	(223)
9.5 鼠标事件	(224)
9.5.1 鼠标位置	(225)
9.5.2 按键状态	(226)
9.5.3 换档状态	(227)
9.6 事务处理	(227)
9.6.1 事务处理机制	(228)
9.6.2 事务处理的规则	(228)
9.6.3 事务处理不支持的命令和函数	(229)
9.6.4 事务处理设计实例	(229)
9.7 本章小结	(230)

第十章 图形处理技术	(231)
10.1 控件的修饰	(231)
10.1.1 图像的装入	(231)
10.1.2 控件的动态调整	(232)
10.2 绘图环境设置	(233)
10.2.1 刻度模式	(233)
10.2.2 背景色和前景色	(233)
10.2.3 画笔的模式、风格和宽度	(233)
10.2.4 填充色和填充风格	(235)
10.3 绘图	(236)
10.3.1 清除	(236)
10.3.2 显示文本	(236)
10.3.3 画点	(236)
10.3.4 直线	(237)
10.3.5 方框	(238)
10.3.6 画圆和椭圆	(238)
10.4 彩色世界	(240)
10.4.1 调色板	(240)
10.4.2 定制颜色调色板	(240)
10.4.3 RGB 函数和颜色值	(241)
10.5 一个基于函数绘图的范例	(242)
10.5.1 数据表结构及其内容	(242)
10.5.2 建立表单集	(244)
10.5.3 建立绘图过程	(246)
10.5.4 为表单集建立新方法	(249)
10.5.5 建立事件	(251)
10.5.6 运行程序绘图	(255)
10.6 程序控制 Ms Graph 图形的显示过程	(255)
10.6.1 图形控制接口	(256)
10.6.2 Ms Graph 图形控制实例程序的设计	(257)
10.7 本章小结	(259)
第十一章 查询和视图设计器、菜单设计器	(261)
11.1 查询和视图设计器	(261)
11.1.1 进入查询和视图设计器	(261)
11.1.2 查询工具栏和视图工具栏	(262)
11.1.3 “查询”菜单和“显示”菜单	(263)
11.1.4 字段页	(265)

11.1.5	联接页	(265)
11.1.6	筛选页	(267)
11.1.7	排序依据页	(267)
11.1.8	分组依据页	(267)
11.1.9	更新条件页	(268)
11.1.10	杂项页	(269)
11.2	菜单设计器和快捷菜单的设计	(269)
11.2.1	菜单系统结构和菜单事件驱动机制	(269)
11.2.2	菜单设计器	(270)
11.2.3	“菜单”项菜单和“显示”项菜单	(272)
11.2.4	用户菜单系统的创建步骤	(274)
11.2.5	菜单系统的总体计划	(274)
11.2.6	设计菜单和子菜单	(275)
11.2.7	为菜单系统指派任务	(275)
11.2.8	建立初始化代码和清理代码	(275)
11.2.9	菜单选项的开关设计	(276)
11.2.10	菜单系统的定制	(276)
11.2.11	创建菜单系统的注意事项	(277)
11.2.12	菜单的保存和菜单程序代码的生成	(277)
11.2.13	菜单系统的控制	(277)
11.2.14	普通菜单、顶级菜单和快捷菜单	(278)
11.2.15	利用数组建立动态快捷菜单	(286)
11.3	本章小结	(290)
第十二章	报表设计器	(291)
12.1	报表设计器	(291)
12.1.1	报表设计器窗口及其分区	(291)
12.1.2	打开报表设计器	(292)
12.1.3	“报表”菜单	(293)
12.1.4	报表控件工具栏	(295)
12.1.5	设置数据环境	(295)
12.1.6	快速制表	(295)
12.2	报表对象设计	(296)
12.2.1	设置报表对象	(296)
12.2.2	设置表格线和圆	(298)
12.2.3	备注字段的处理	(299)
12.3	页面设置	(299)
12.3.1	页面设置	(299)
12.3.2	报表每页打印的记录数控制	(299)

12.3.3	报表分组	(300)
12.3.4	多列蛇型栏报告的生成	(301)
12.3.5	如何添加“承接上页”标记	(301)
12.3.6	空行不打印	(301)
12.3.7	建立报表样板	(302)
12.4	报表变量的使用	(302)
12.5	报表预览和输出	(304)
12.5.1	报表预览和预览工具栏	(304)
12.5.2	报表程序的运行	(304)
12.5.3	报表的满页打印和有效区域打印	(305)
12.6	正确设置打印环境	(305)
12.7	本章小结	(306)
第十三章	向导和项目管理	(307)
13.1	向导	(307)
13.1.1	表单向导	(307)
13.1.2	一对多表单向导及表单向导中的“BUG”	(311)
13.1.3	表向导	(313)
13.1.4	报表向导	(313)
13.1.5	分组总计报表向导	(314)
13.1.6	一对多报表向导	(314)
13.1.7	标签向导	(315)
13.1.8	查询向导	(315)
13.1.9	图形向导	(316)
13.1.10	文档向导和文档向导的定制	(316)
13.1.11	安装向导	(320)
13.1.12	应用程序向导	(326)
13.1.13	WWW 搜索页向导	(330)
13.2	项目管理器	(335)
13.2.1	进入项目管理器	(335)
13.2.2	“项目”菜单	(335)
13.2.3	项目管理器中的按钮使用	(337)
13.2.4	项目的扩展或压缩显示及双击操作	(338)
13.2.5	项目管理器窗口的压缩和扩展显示及码放	(338)
13.3	本章小结	(339)
第十四章	定制联机帮助系统	(340)
14.1	帮助系统及其结构	(340)
14.1.1	帮助驱动程序和帮助文件	(340)

14.1.2	帮助窗口的结构	(340)
14.1.3	帮助系统提供的帮助方式	(344)
14.2	帮助系统的设计方法	(344)
14.2.1	帮助系统的需求和规范	(344)
14.2.2	确定帮助系统的层次结构	(345)
14.2.3	帮助主题文件的编写	(345)
14.2.4	帮助文件的生成和调用	(347)
14.3	编制帮助主题文件脚本	(347)
14.3.1	选择帮助主题文件编辑器	(347)
14.3.2	帮助主题文件脚本编码	(348)
14.4	构筑一个好的帮助项目文件	(354)
14.4.1	帮助项目文件的格式	(354)
14.4.2	帮助项目文件的节的使用	(355)
14.5	帮助副窗口和超图的使用	(359)
14.5.1	帮助副窗口的使用	(359)
14.5.2	定义副窗口	(360)
14.5.3	引用副窗口	(360)
14.5.4	超图的创建	(361)
14.5.5	超图的使用	(362)
14.6	帮助宏指令的使用	(362)
14.6.1	指令的引用方式和编辑规则	(362)
14.6.2	常用的帮助宏指令	(364)
14.7	建立 VFP 5.0 应用的上下文敏感帮助	(370)
14.7.1	帮助主题的上下文号及其分配	(370)
14.7.2	在 VFP 5.0 中实现上下文敏感帮助	(370)
14.8	用 Word 宏建立帮助主题文件编辑工具栏	(374)
14.8.1	Word 的遗憾	(375)
14.8.2	用宏提供解决方案	(376)
14.8.3	帮助编辑工具栏的加载	(379)
14.9	本章小结	(381)

附录一 Visual FoxPro 系统容量 (382)

表文件及索引文件	(382)
字段的特征	(382)
内存变量与数组	(382)
程序和过程文件	(383)
报表设计器的容量	(383)
其他的容量	(383)

附录二 数据表文件的结构.....	(384)
表头记录的结构.....	(384)
字段子记录结构.....	(385)
附录三 表文件的表结构.....	(386)
数据库文件(.DBC)的表结构	(386)
表单文件(SCX)和可视类库(VCX)文件的表结构	(386)
菜单文件(MNX)的表结构	(387)
项目文件(PJX)的表结构	(388)
报表文件(FRX)和标签文件(LBL)的表结构.....	(389)
备注文件的结构 (.FPT)	(391)
备注头记录.....	(391)
备注块标头与备注文本.....	(391)
索引文件的结构.....	(391)
单索引文件的结构(.IDX)	(391)
压缩索引文件的结构.....	(392)
复合索引文件的结构 (.CDX).....	(393)
宏文件的格式 (.FKY).....	(393)

第一章 Visual FoxPro 5.0 概述

Visual FoxPro 5.0 产品,是 Microsoft 公司于 1997 年正式推出的 FoxPro 新产品。该产品是第一个真正与 Windows 95 /NT 兼容的全 32 位面向对象的 FoxPro 快速应用程序开发环境。作为 Microsoft Visual Studio 系列开发工具(Visual Basic 5.0, Visual C++ 5.0, Visual J++)的一员,Visual FoxPro 5.0 增加了许多借鉴于 Visual Basic 和 Visual C++ 的新功能。

与以前版本比较,Visual FoxPro 5.0 具有了许多新的特性、功能和工具。关系数据库技术得到了更为系统、完善的实现,表级和字段级数据完整性验证机制有力地保证了数据库的完整和安全;尽管它仍保持了结构化过程设计的能力,但这仅仅是为了与原有应用软件的兼容而遗留下来的过渡功能,面向对象编程和可视编程已成为它极力推荐的编程方法,并为此提供了许多新的增强的软件开发工具。

现在的 Visual FoxPro 5.0 版本,还带来了更强大的功能,可以在应用程序或数据库开发的任何一个领域中给我们提供帮助。这些改进体现在性能、系统资源利用以及设计环境上。

1.1 完善的数据库引擎

Visual FoxPro 5.0 仍然是一个关系型数据库系统。但较 Visual FoxPro 3.0 及 FoxPro 2.6 以前版本,它的数据库引擎技术更趋完善。

1.1.1 引入了数据词典

Visual FoxPro 5.0 新增了一个数据词典的特性。数据库的数据词典特性便于用来设计与字段层次相关联的对象所属的对象类(可以是自定义类),并且可以让用户通过快速拖曳的方式来直接设计所需的用户界面中的编辑对象。

对数据表中的字段,Visual FoxPro 5.0 增加了两个用于显示控制的属性:Format 属性和 InputMask 属性,它们将在表单设计器中使用拖曳技术来设计编辑字段的对象时,控制该对象的显示属性如 Format,InputMask 等属性的值;同时可以设定字段类型所对应的对象类,在表单设计时,它将决定通过拖曳设定该字段时对应的对象类。

1.1.2 增强的数据库管理

数据库容器允许几个用户在同—个数据库中同时创建或修改对象。按需刷新的特性允许更新数据库或视图。数据库设计器具有查找及重排的特性,允许迅速更改数据库中对象的视图,并可以使用键盘访问项目管理器及项目管理器中的图标,数据库设计器也能帮助我们迅速区分对象。

现在,对于和规则相冲突的值,数据库引擎支持对它们的更改。保守式缓冲可以更有效地处理记录的锁定。NULL 现在已经是一个可标识的关键字,可以代替“. NULL.”使用。

同时, Visual FoxPro 5.0 改进了的表设计器, 将更加易于进行字段的索引设置。

1.1.3 加强数据完整性验证机制

在 Visual FoxPro 5.0 中, 引进和完善了关系数据库的三类完整性: 实体完整性、参照完整性和用户自定义完整性。实体完整性规则, 要求数据表中记录的主键值不能是空值, 如果出现空值, 那么主键值就起不了唯一标识记录的作用; 参照完整性规则, 要求不允许在相关数据表中引用不存在的记录; 用户自定义的完整性规则, 是针对某一具体数据表的约束条件, 由应用环境决定, 它反映某一具体应用所涉及的数据必须满足的语义要求。

前两类完整性由数据库本身所固有。系统同时提供了用户自定义完整性的机制, 以便使用统一的、系统的方法处理它们, 而不需要由某个应用程序担负这一功能。

1.1.4 扩充的数据类型

Visual FoxPro 5.0 能支持的数据类型有 12 种: 货币金额型 (Currency)、日期时间型 (DateTime)、双精度型 (Double)、二进制字符型 (Binary Character)、二进制备注型 (Binary Character) 等五个字段类型, 大大地扩充了系统的计算和存储能力。

1.1.5 数据环境、表和关联成为对象

在表单设计器中, 一个表单或表单集可以具有自己的数据环境, 包括数据库中一些数据表、数据表间的关联。数据环境、数据表和数据表间的关联, 将被当作对象看待, 它们可以具有自己的属性、事件和方法, 因而可以在代码中对数据环境、数据表及其关联进行精确的控制。譬如, 可以在代码中控制: 自动打开或关闭数据表、向数据环境添加和删除数据表、建立数据表间关联等等操作。

1.1.6 使用游离视图增强连接能力

有时, 可能要经常显示、收集或修改与主数据库独立的数据。通过在 Visual FoxPro 5.0 中创建游离视图, 我们就可以使用与主数据库联接的视图, 并为脱机使用创建一个数据子集。

如果要脱机工作, 可以直接使用该视图, 或创建一个应用程序来完成。完成后, 可将视图中的修改升迁到主数据库中。当升迁游离视图时, Visual FoxPro 5.0 对游离视图和原始数据间的改变进行协调管理。

1.2 引进了先进的程序设计技术

正如 Visual Basic 5.0, Visual C++ 5.0 等产品一样, 在 Visual FoxPro 5.0 系统中, 也全面使用了面向对象编程和可视编程技术。

1.2.1 面向对象编程

在 FoxPro 2.6 以前版本中, 尽管局部地使用了面向对象 (Object-Oriented) 的编程技术, 但是, 从总体来说, 它仍旧脱不开面向过程的结构化设计框框。随着应用的日趋复杂, 由于面向过程的设计方法很难做到模块及数据的真正独立, 其开发和维护越来越困难, 因而也越来越不

能适应需要。因此, Visual FoxPro 5.0 在 Visual FoxPro 3.0 的基础上, 开始全面采用了面向对象编程的设计思想。

在 Visual FoxPro 5.0 中, 将现实世界的实体, 都抽象成一个一个封装好的对象类, 如表单 (Form)、各种屏幕和报表控件 (Control) 等, 它们具有属性 (Property)、事件 (Event) 和方法 (Method)。这些对象类具有封装性 (Capsulation)、继承性 (Inheritance) 和多态性 (Polymorphism)。对象类的封装性, 使有关该对象类的属性、事件和方法, 全部封装到了对象类里面, 对外界是完全不透明的, 对象类完全拥有它自己的属性。程序设计者根本不需要了解对象类中的技术代码, 也无法对它们加以控制和干预, 而只需重复调用对象即可。对象类的继承性, 则允许在已有的对象类的基础上构造新的对象类, 即由一个对象类获取另一个对象类的过程。对象类的多态性允许程序开发者通过向一个对象发送消息来从事一系列的活动, 而不必关心对象内部是如何解释这个活动以及该对象如何实现这些活动的, 而关心的仅仅是这个活动对该对象产生的作用。

我们可以通过设置对象的属性、控制对象的事件处理、调用对象的方法, 即可按照我们的意愿, 完成各式各样的具体任务。由于各种对象具有十分丰富的属性、事件和方法, 因而我们开发应用的控制能力更强, 完全可以做到对对象的精确控制。譬如说, 可以利用文本框或编辑框控件的 KeyPress 事件, 对编辑域中录入的一个一个字符进行控制, 从中可以剔除非合法键入和检测功能字符, 并进行相应的处理; 我们也可以精确地控制鼠标按键的按下、拖曳、释放等事件, 甚至可以精确地控制当拖曳鼠标进入或退出某一个控件或区域时应发生什么事情; 在程序的运行过程中, 控件的尺寸缩放和位置调整, 也成为轻而易举的事情。然而, 这些能力在 FoxPro 2.6 以前版本中一直是一个奢望。

面向对象编程允许利用许多预先设计好的封装了代码和数据的对象, 迅速构筑一个大的应用软件。对象把代码和数据融为一个整体, 从而能完整地描述现实世界某一对象的全貌。已封装的对象可以重复利用, 软件生产变得相对容易, 程序质量也变得更加可靠。它彻底地改变了传统的编程方法, 大大提高了软件的开发效率。

1.2.2 可视化编程

如果我们使用过 FoxPro 2.6 以前版本, 一定熟悉它们提供的屏幕设计器 (Screen Designer) 工具。在屏幕设计器工具中, 已为屏幕构造提供了有限的可视编程能力。我们可以在一个窗口上随意放置屏幕控件, 并通过激活一个对话框选择对控件的 VALID 和 WHEN 等有限的几个子句进行控制, 其控制能力相当有限。

在 Visual FoxPro 5.0 中, 则提供了一套完整的可视化编程环境。它放弃了老版本的屏幕设计器, 取而代之的是一个表单设计器 (Form Designer)。表单设计器提供了一个属性 (Properties) 窗口和一个事件、方法代码 (Code) 窗口, 并提供了众多的工具栏, 以便对对象的属性进行精细的可视控制。所有对对象的属性调整, 都立即反映到了显示的控件上。由于对象提供的属性很多, 因此可以控制的属性也很多。譬如, 可以利用色彩工具栏可视地控制命令按钮上的字符色彩, 可以产生控件的立体效果等等。对象的事件和方法的代码编辑, 可以直接在代码编辑窗口中通过简单地挑选不同的对象、不同的事件或方法, 来实现对不同对象的不同事件的代码编辑。而且, 在当前活动对象发生了变化时, 其属性窗口和代码窗口都自动跟着发生相应的转换, 毋需再人工在不同对象之间进行切换。