

Microsoft Visual Studio 98 系列丛书

李 东 陈 群
蒋志浩 等编著

Microsoft

中文 Visual FoxPro 6.0
程序设计基础

北京航空航天大学出版社

Microsoft Visual Studio 98 系列丛书

中文 Visual FoxPro 6.0 程序设计基础

李 东 陈 群 蒋志浩 等编著

北京航空航天大学出版社

T13

内 容 简 介

本书是一本介绍中文 Visual FoxPro 6.0 编程基础知识的参考书。全书共分十九章, 内容包括: 中文 Visual FoxPro 6.0 简介; 安装与配置、常用系统工具、开发环境、应用程序开发技术; 用项目管理器编写应用程序; 用 VFP6.0 程序设计语言; 用 SQL 查询语言、用面向对象的程序设计、事件模型、数据库和表、表处理技术、视图操作、表单、使用控件、菜单和工具栏建立数据环境; 还用数据库设计一个应用程序及实例。

本书适合于利用 Visual FoxPro 进行程序设计的专业或非专业初、中级开发人员参考使用, 同时也可用作大、中专院校和一些计算机培训班的参考书。

图书在版编目(CIP)数据

中文 Visual FoxPro 6.0 程序设计基础/李东等编著. -
北京: 北京航空航天大学出版社, 1999. 1
ISBN 7-81012-850-7

I. 中... II. 李... III. 关系数据库-数据库管理系统. F
oxPro6.0-程序设计 IV. TP311

中国版本图书馆(CIP)数据核字(98)第 32525 号

中文 Visual FoxPro 6.0 程序设计基础

李 东 陈 群 蒋志浩 等编著

责任编辑: 徐友泉

封面设计: 李宝田

北京航空航天大学出版社出版发行

北京市学院路 37 号, 邮编 100083 发行部电话: 010-82317024

<http://www.buaapress.cn.net>

E-mail: pressell@publica.hj.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: 21.75 字数: 557 千字

1999 年 1 月第 1 版 1999 年 1 月第 1 次印刷 印数: 5000 册

ISBN 7-81012-850-7/TP·323 定价: 27.00 元

前 言

Microsoft Visual FoxPro 6.0 for Windows (简称 VFP6.0) 是 Microsoft 公司推出的最新一代全 32 位、面向对象的数据库开发管理工具。它的发展及使用不仅大大简化了数据库的管理,使组织数据、创建用户应用程序等工作更加快捷,而且还提供了功能强大的面向对象编程的工具、OLE 支持及 Web 服务能力,使开发应用程序的功能更加完备。

Visual FoxPro 6.0 继承了以往所有版本数据库管理系统的功能,并扩展了对应用程序的管理和在 Internet 上发布用户数据的功能,使得开发数据库工具更加完善与快捷,从而成为当今功能十分强大、性能更为可靠的数据库管理系统。

Visual FoxPro 6.0 同以前的数据库管理系统相比,具有更快速、更有效、更灵活的突出特点。它能够迅速而简单地建立用户的数据库,从而方便地使用和管理数据:不仅支持客户机/服务器 (C/S) 结构,而且具有与其他软件 (如 Excel, Word) 共享和交换数据的能力。

Visual FoxPro 6.0 提供对象和事件处理模式,利用面向对象编程 (OOP) 的优势使用户能够快速建立和修改应用程序。Visual FoxPro 6.0 与以前版本的 FoxPro 完全兼容,即用户以前的应用程序不经修改可直接在 Visual FoxPro 6.0 上运行。

本书内容全面,循序渐进,适合对 Visual FoxPro 6.0 感兴趣的初、中级读者参考使用。

除署名外,参加本书编写的还有程洁、刘闯、张红、赵丽华、邹文彪、郑文龙、李兵。本书由曹春光策划,武力、冯海涛统稿,农军、王海宁审校。本书录入排版工作由朱志岩、张文玲完成。在此,对参与本书编写、录排的所有同志深表感谢。

编 者

1998 年 12 月

目 录

第一章 中文 Visual FoxPro 6.0 简介	1
1.1 Xbase 系列数据库系统简介	1
1.2 中文 Visual FoxPro 6.0 新特点	2
1.2.1 增强的网络服务功能	2
1.2.2 更多更好的向导	2
1.2.3 OLE 拖放 (Drag and Drop) 技术	3
1.2.4 2 000 年时间的支持	3
1.2.5 增强的应用程序框架	3
1.2.6 Automation 服务程序的改进	4
第二章 中文 Visual FoxPro 6.0 安装与配置	5
2.1 中文 Visual FoxPro 6.0 安装步骤	5
2.1.1 系统要求	5
2.1.2 安 装	5
2.1.3 定制系统	10
2.1.4 常见问题	13
2.2 进 入	14
2.3 退 出	14
2.4 定制开发环境	14
2.4.1 系统设置	14
2.4.2 环境的交互式设置	15
2.4.3 启动时设置配置选项	18
2.4.4 配置工具栏	22
2.4.5 设置编辑器选项	23
2.4.6 恢复环境	24
2.5 系统优化	25
2.5.1 配置操作环境	25
2.5.2 优化系统的启动速度	27
2.5.3 在多用户环境中优化 Visual FoxPro 6.0	28
第三章 常用系统工具	30
3.1 系统工具简介	30
3.3 项目管理器	31
3.3.1 项目管理器功能简介	31
3.3.2 项目管理器的组成	33
3.3.3 用项目管理器创建和修改文件	34
3.3.4 用项目管理器添加和删除文件	34
3.3.5 项目间的文件共享	34
3.3.6 浏览表的内容	35
3.3.7 项目管理器的定制	35
3.4 使用设计器	36
3.4.1 用设计器定制应用程序	36

3.4.2	使用数据库设计器设计数据库.....	37
3.4.3	使用表设计器创建表及索引.....	38
3.4.4	用查询设计器创建查询.....	40
3.4.5	用视图设计器创建视图.....	41
3.4.6	创建菜单栏和快捷菜单.....	42
3.5	使用工具栏.....	42
3.6	使用向导.....	43
3.6.1	不同向导的作用.....	43
3.6.2	启动向导.....	44
3.6.3	定制向导操作.....	45
3.6.4	保存向导结果.....	45
3.6.5	修改用向导创建的项.....	45
3.7	使用生成器.....	45
3.7.1	生成器的作用.....	46
3.7.2	向表单增加控件.....	47
3.8	用 Visual FoxPro 6.0 的控件创建界面.....	48
3.8.1	创建一对多表单.....	48
3.8.2	使表单外观专业化.....	50
3.8.3	设置控件属性.....	50
3.8.4	定制表单设计器.....	51
3.8.5	从类库中添加控件.....	51
3.8.6	扩展类库.....	52
3.9	Visual FoxPro 6.0 系统工具应用实例.....	53
3.9.1	创建报表.....	53
3.9.2	创建帮助系统.....	54
第四章	中文 Visual FoxPro 6.0 开发环境.....	55
4.1	VFP 6.0 集成开发环境.....	55
4.2	VFP 6.0 主窗口.....	56
4.3	系统菜单.....	56
4.3.1	“文件”菜单.....	56
4.3.2	“编辑”菜单.....	58
4.3.3	“显示”菜单.....	59
4.3.4	“格式”菜单.....	60
4.3.5	“工具”菜单.....	61
4.3.6	“程序”菜单.....	62
4.3.7	“窗口”菜单.....	62
4.3.8	“帮助”菜单.....	63
4.4	Visual FoxPro 6.0 的工具栏.....	63
4.4.1	定制工具栏的种类.....	64
4.4.2	定制每个工具栏中的项目.....	73
4.4.3	定制工具栏的显示方式.....	74
第五章	应用程序开发技术.....	76
5.1	应用程序总体设计.....	76
5.2	创建应用程序的过程.....	77
5.3	开始开发工作.....	77
5.4	使用项目管理器.....	77

5.5 创建数据库	78
5.6 创建类	78
5.7 设计访问信息的方法	79
5.8 测试和调试	79
第六章 用项目管理器编写应用程序	80
6.1 项目管理器的主要功能	80
6.2 使用项目管理器	81
6.2.1 展开/折叠按钮	81
6.2.2 项目管理器按钮	82
6.2.3 使用项目管理器	83
6.3 项目管理器的定制	84
6.4 用项目管理器快速开发程序	84
6.4.1 新建一个项目	84
6.4.2 用“新建文件”创建一个项目	86
6.4.3 用“向导”创建一个项目	86
6.5 设计应用程序的一般过程	89
6.5.1 构造应用程序框架	89
6.5.2 向项目中添加文件	91
6.5.3 引用可修改的文件	91
6.5.4 运行应用程序	91
第七章 VFP6.0 程序设计语言	92
7.1 数据类型和字段类型	92
7.1.1 主要数据类型	92
7.1.2 字段类型	93
7.2 数据存储方式	94
7.2.1 变 量	94
7.2.2 常 量	94
7.2.3 数 组	95
7.2.4 字 段	95
7.2.5 记 录	95
7.2.6 对 象	95
7.2.7 操作符	95
7.2.8 创建表达式	96
7.2.9 宏替换	97
7.2.10 NULL 值	97
7.3 VFP6.0 的基本语法结构	98
7.3.1 基本的赋值语句以及表达式	98
7.3.2 IF ... ELSE ... ENDIF	99
7.3.3 DO CASE ... ENDCASE	100
7.3.4 SCAN ... ENDSCAN	101
7.3.5 FOR ... ENDFOR	102
7.3.6 DO WHILE ... ENDDO	102
7.3.7 函 数	103
7.4 过程和用户定义函数	103
7.4.1 过 程	103
7.4.2 函 数	104

第八章 SQL 查询语言	106
8.1 SQL 语言	106
8.2 在 VFP 中进行查询	106
8.3 SELECT 命令	107
8.4 其他常用 SQL 命令	113
8.4.1 ALTER TABLE 命令	113
8.4.2 CREATE CURSOR 命令	116
8.4.3 CREATE TABLE 命令	117
8.4.4 DELETE 命令	119
8.4.5 INSERT 命令	120
8.4.6 UPDATE 命令	120
8.5 SQL 查询示例	121
第九章 面向对象的程序设计	124
9.1 面向对象编程基础	124
9.1.1 类	124
9.1.2 对象与方法	125
9.1.3 封装、继承和多态性	125
9.2 Visual FoxPro 中的对象和方法	126
9.3 Visual FoxPro 中的类	128
9.3.1 类的概念	128
9.3.2 可视类和不可视类	130
9.3.2 Visual FoxPro 类的使用	132
第十章 VFP6.0 的事件模型	136
10.1 Visual FoxPro 中的核心事件	136
10.2 追踪事件序列	139
10.3 监视事件	139
10.4 为事件指定代码	143
第十一章 数据库和表	145
11.1 数据库和表的概念	145
11.2 关系概念	146
11.2.1 一对一关系	146
11.2.2 一对多关系	146
11.2.3 多对多关系	146
11.3 创建数据库一般步骤	147
11.3.1 使用数据字典功能	147
11.3.2 字段级限制	147
11.3.3 记录级限制	147
11.3.4 创建触发器	147
11.3.5 创建存储过程	148
11.3.6 创建本地视图	148
11.3.7 定义连接	148
11.3.8 参照完整性的建立	148
11.3.9 创建永久关系	150
11.3.10 确定表字段的规则	150

11.4	数据库设计方法.....	150
11.5	表的设计方法.....	154
11.5.1	通过表向导设计表	154
11.5.2	通过表设计器设计表.....	158
11.6	建立索引	162
11.6.1	选择索引类型	162
11.6.2	设置主索引、候选索引.....	163
11.6.3	设置普通索引	163
第十二章	表处理技术	164
12.1	表的处理	164
12.1.1	设计数据库和自由表.....	164
12.1.2	创建数据库表	165
12.1.3	创建自由表	165
12.1.4	命名表	165
12.1.5	重命名表	166
12.1.7	删除自由表	166
12.1.8	复制表	167
12.1.9	复制和编辑表结构	167
12.1.10	将表另存为 HTML 文件	167
12.1.11	命名字段	168
12.1.12	选择数据类型	169
12.1.13	创建字段的默认值	170
12.1.14	控制字段显示	172
12.1.15	实施商业规则	173
12.1.16	使用触发器	176
12.1.17	创建触发器	176
12.1.18	修改触发器	177
12.1.19	使用触发器建立参照完整性.....	177
12.1.20	修改表结构	177
12.1.21	添加字段	178
12.1.22	删除字段	178
12.1.23	重新命名字段	178
12.1.24	设置或更改字段级规则或表规则.....	178
12.1.25	设置或更改默认值	179
12.2	记录的处理	179
12.2.1	添加记录	179
12.2.2	从其他表中追加记录.....	180
12.2.3	采用浏览方式添加数据.....	180
12.2.4	向表中输入数据	180
12.2.5	在表中编辑记录	180
12.2.6	向表中添加图形	180
12.2.7	在字段中输入 NULL 值.....	181
12.2.8	删除记录	181
12.2.9	对记录作删除标记	181
12.2.10	检索具有删除标记的记录.....	181
12.2.11	如何撤销记录的删除标记.....	181

12.2.12	删除带有删除标记的记录.....	182
12.2.13	保存空间	182
12.2.14	从表中移去所有记录.....	182
12.3	表的索引	182
12.3.1	创建一个索引	183
12.3.2	创建索引文件	183
12.3.3	查看索引信息	184
12.3.4	控制重复值	184
12.3.5	创建多个索引	185
12.3.6	控制访问记录的顺序.....	186
12.3.7	在运行时刻设置记录顺序.....	186
12.3.8	使用非结构 (.cdx) 索引.....	189
12.3.9	删除索引	190
12.3.10	使用表达式进行索引.....	190
12.3.11	筛选数据	193
12.3.12	高效地使用索引	195
12.4	使用多个表	195
12.4.1	使用数据工作期	195
12.4.2	查看工作区	196
12.4.3	如何在工作区中打开表.....	196
12.4.4	在工作区中关闭表	197
12.4.5	引用工作区	197
12.4.6	使用表别名	197
12.4.7	设置表间的临时关系.....	198
12.4.8	临时相关表	198
12.4.9	用索引设置永久关系.....	200
第十三章	视图操作	201
13.1	创建视图	201
13.1.1	创建本地视图	201
13.1.2	用已有的 SQL SELECT 语句创建视图.....	202
13.1.3	修改视图	202
13.1.4	重新命名视图	202
13.1.5	删除视图	203
13.1.6	创建多表视图	203
13.1.7	在结果中包含不匹配记录.....	204
13.1.8	访问远程数据	205
13.1.9	创建远程视图	207
13.2	使用视图	207
13.2.1	显示视图	207
13.2.2	限制视图的取值范围.....	208
13.3	打开视图	210
13.3.1	在多个工作区中打开一个视图.....	211
13.3.2	不下载数据再次使用视图.....	211
13.3.3	显示视图结构	212
13.3.4	创建视图索引	212
13.4	视图属性和连接属性.....	212

13.5 在视图中更新数据.....	215
13.5.1 更新视图的基本操作.....	215
13.5.2 在视图中更新多个表.....	216
13.5.3 用数据字典定制视图.....	218
13.5.4 为视图字段创建默认值.....	218
13.5.5 在视图的字段和数据行上创建规则.....	218
13.6 在视图中集成数据.....	219
13.6.1 在视图中集成本地数据和远程数据.....	219
13.6.2 在视图中更新本地数据与远程数据.....	220
13.7 处理自由数据.....	220
13.7.1 创建游离视图.....	220
13.7.2 显示和修改自由数据.....	221
13.8 优化视图.....	223
13.8.1 控制获取信息量的大小.....	223
13.8.2 控制备注字段的获取.....	224
13.8.3 设置下载的最大记录数.....	224
13.8.4 优化过滤器和连接.....	224
13.8.5 共享多个远程视图的连接.....	226
第十四章 表 单.....	227
14.1 设计表单.....	227
14.1.1 使用表单设计器.....	228
14.1.2 使用表单向导.....	230
14.2 保存和运行表单.....	235
14.3 自定义表单.....	236
14.3.1 改变表单中文本的字体和大小.....	236
14.3.2 向表单中添加线条.....	236
14.3.3 向表单中添加形状.....	237
14.3.4 向表单中添加图形.....	237
14.3.5 设置表单的前景和背景颜色.....	238
14.4 添加控件.....	239
14.4.1 调整控件的大小和外形.....	240
14.4.2 表单的布局.....	240
14.5 创建表单集.....	242
14.6 修改表单属性.....	243
第十五章 使用控件.....	244
15.1 深入了解控件和数据.....	244
15.2 根据任务选择合适的控件.....	245
15.2.1 提供一组预先设定的选择.....	245
15.2.2 接受预先不能确定的输入.....	256
15.2.3 接受指定范围的数值输入.....	260
15.2.4 允许特定操作.....	261
15.2.5 在给定时间间隔执行指定操作.....	263
15.2.6 显示信息.....	265
15.2.7 增强控件显示.....	267
15.2.8 操作多行数据.....	267
15.3 使控件易于使用.....	273

15.3.1	设置访问键	273
15.3.2	设置工具提示文本	274
15.3.3	启用和废止控件	275
15.3.4	允许用户拖放	275
15.4	扩展表单	278
15.4.1	使用页框	278
15.4.2	ActiveX 控件	281
15.4.3	ActiveX 绑定控件	281
第十六章	菜单和工具栏	282
16.1	设计菜单系统	282
16.1.1	规划菜单系统	282
16.1.2	创建图形化菜单界面	283
16.1.3	菜单的进一步设计	287
16.1.4	实现菜单的具体功能	290
16.1.5	预览菜单	294
16.1.6	生成菜单程序	295
16.1.7	菜单代码示例	296
16.1.8	菜单的常规选项	299
16.1.9	菜单选项	299
16.2	设计工具栏	300
16.2.1	创建工具栏	300
16.2.2	将工具栏连接到表单	303
16.2.3	工具栏的特点	306
16.3	调整用户设计的菜单和工具栏	306
第十七章	建立数据环境	307
17.1	设置数据环境	307
17.2	添加表或视图	308
17.3	将表从数据环境中移去	309
17.4	为数据环境设置关系	309
17.5	在数据环境设计器中编辑关系	311
17.6	向表单中添加表字段	311
第十八章	数据库设计实例	314
18.1	设计表结构	314
18.2	设计数据库	317
18.3	设计各表间的关系	319
18.3.1	建立关系	319
18.3.2	编辑关系	320
18.3.3	修改关系	320
18.4	建立参照完整性	322
18.4.1	更新规则	322
18.4.2	删除规则	322
18.4.3	插入规则	323
18.5	浏览数据库和表	324
第十九章	设计一个小应用程序	326

19.1	设计目标	326
19.2	设计表单的界面.....	327
19.3	建立数据环境.....	328
19.4	向表单添加控制字段.....	329
19.5	调整表单	329
19.6	添加专用代码.....	330

第一章 中文Visual FoxPro 6.0简介

Visual FoxPro 6.0 (简称 VFP 6.0) 是一个完全的面向对象程序设计技术与传统的过程化程序设计模式相结合的开发环境, 它建立在事件驱动模型的基础之上, 给程序的开发提供了极大的灵活性。要真正掌握 VFP 6.0, 并使用它开发专业级的应用程序, 就必须透彻地理解面向对象的程序设计思想、事件驱动模型, 并把它们完美、紧密地结合起来, 只有这样才能创建真正高效、功能强大、灵活性强的 Visual FoxPro 6.0 应用程序。

本章的主要内容是 Visual FoxPro 6.0 中文版简介、特点、安装。在安装部分涉及到系统运行环境、系统定制以及安装方法等, 除此之外, 还讲述了在安装中碰到的各种问题以及解决办法。

对于 Visual FoxPro 6.0 中文版来说, 系统的配置及优化是一个重要的话题, 也是影响系统性能的关键因素。本章详细讲述了系统的配置及优化措施。

系统配置有几种方式: Windows 系统注册表、中文版配置文件、“选项”对话框和 SET 命令。

系统的优化包括以下几个方面: 启动速度、关键的 SET 命令及多用户环境下的优化。其中, 多用户环境下的优化包括有效地管理临时文件和控制文件的共享方式。

另外, 本章还讲述了 Visual FoxPro 6.0 中文版的各个启动开关以及卸载方法。

1.1 Xbase系列数据库系统简介

数据库应用系统是一种最常见的应用系统, 也是信息管理领域的支柱。Xbase 系列数据库系统开发工具始终是在同阶段中人们使用最广泛、最易使用的开发工具之一。从 dBaseI, dBaseII, dBaseIII, 到 Foxbase, 随后又引入 FoxPro 2.x 版本 (包括 FoxPro 2.x for DOS 和 FoxPro 2.x for Windows), 随着 GUI 用户界面开发工具的不断更新换代, 继而又过渡到 Visual FoxPro 系列 (包括 Visual FoxPro 3.0, Visual FoxPro 5.0 和 Visual FoxPro 6.0 版本)。其中, 每次产品的更新都给了人们很多的惊喜和意外。

从 dBase 系列到 Foxbase 的过渡, 主要体现在数据库管理系统的设计功能上有了很大的进步, 它很明显地使数据库的开发更加方便、简洁。但是, 在设计用户界面问题上, 虽然对功能作了很大的改进, 但手工制作的成分还是很大, 致使系统生成的功能不强。

FoxPro 2.x 引入了屏幕生成器的概念, 部分体现了所想即所得的思想, 系统已经能够自动生成某种程度的程序代码, 但自动程度不够高, 而且很不灵活。

随着 Windows 风格用户界面的逐步统一, 编程的可视化使开发工具又面临一大挑战。挑战的结果, 出现了很多的开发工具, 如 PowerBuilder, Access 以及 Visual FoxPro 系列产品等。Microsoft 的 Visual 系列开发工具, 以它自己独到的特点和优势赢得了越来越多的开发者。它彻底实现了所见即所得、所想即所得的思想, 完美地实现 GUI 用户界面用户可以自由定制的功能。人们可以不去为制作 Windows 风格用户界面而耗费很大的精力, 而把精力集中在核心代码的实现上。它采用了事件驱动、面向对象的程序设计方式, 使得数据库信息管理

系统的开发周期变得非常地短，人们投入的人力只是传统开发中极少的一部分。

1.2 中文 Visual FoxPro 6.0 新特点

Microsoft Visual FoxPro 6.0 是 Microsoft 公司推出的最新一代全 32 位、面向对象的数据库开发管理工具。比起它的早期版本，Visual FoxPro 6.0 支持更强的网络功能和应用程序管理能力，主要表现在以下方面。

1.2.1 增强的网络服务功能

Visual FoxPro 6.0 提供了更多、更好的工具使用户能够轻松地将自己的应用程序扩展到 Internet 网页上去。

1. 生成 HTML 文件

新增的“Web 发布”向导可以让用户按照不同的使用要求将数据库表或自由表转换成 HTML 文件，并直接公布在相应的 Web 浏览器上。用户在指定生成的 HTML 文件是动态还是静态时，只需要在定义 Web 页面布局及样式时设置相应的选项。同时，也可以在“Web 发布”向导中为生成的 Web 页面创建脚本文件 (*.prg)，这样，在 Visual FoxPro 环境中选择“程序”菜单的“运行”命令并执行相应的脚本文件后，就可以在 Web 浏览器上查看 Web 页面了。

用户打开一个数据库表或自由表后，可以通过选择“文件”菜单的“另存为 HTML”命令直接将数据转化为 HTML 文件，并公布在 Web 浏览器上。

注意：要想通过“另存为 HTML”命令转换一个表，该表必须处于“浏览”状态。

2. Active 文档

Active 文档是一种在 Intranet 上使用用户 Visual FoxPro 应用程序的方法。用户可以通过 Active 文档直接在 Microsoft 的 Internet 网页浏览器上运行 Visual FoxPro 创建的表单，其中包括 Visual FoxPro 的源代码。

在 Visual Basic 中，Active 文档仅仅对表单作了极小的修改以支持 Internet 的浏览特性，而在 Visual FoxPro 中，ActiveDoc 类是一种不可视类，它更像运行于浏览器中的 Visual FoxPro 表单控件所需的应用程序对象。Active 文档在 Internet Explorer 标准用户界面中增加了一些新的属性设置，如旋转条 (ScrollBar)、标题栏 (TitleBar) 等。

注意：在 Internet Explorer 中运行的 Visual FoxPro Active 文档只是一个 Visual FoxPro 的应用程序（具有扩展名 .app），因此，在 Internet 上运行 Active 文档，必须要 Visual FoxPro 的运行文件存在，即必须安装 Visual FoxPro。基于这样的要求，Active 文档更适合在 Intranet 上使用。

在 Visual FoxPro 6.0 的操作环境中，可以在 Internet 浏览器上直接使用“打开”命令，执行 Visual FoxPro 生成的应用程序文件 (.app)，并在浏览器上进行浏览。

1.2.2 更多更好的向导

除了上面提到的“Web 发布”向导外，Visual FoxPro 6.0 提供更多、功能更强的向导。例如，新版本的 Visual FoxPro 取消了分组/总计报表向导，取而代之的是在“一对一报表”和“一对多报表”向导中增添了分组和总计的功能选项，从而简化并加强了报表的设计。

Visual FoxPro 6.0 提供了两种应用程序向导。除了 5.0 版本的应用程序向导外，在新的“应用程序向导”中，“应用程序生成器”的使用简化了应用程序的建立，使用户只须输入应用程序的名称并设置相应的信息，即可建立新的应用程序框架，并在应用程序生成器中完成应用程序的设计。5.0 版本中的“Internet 搜索向导”在新版本中变为“WWW 搜索页向导”，其功能基本一样。

新的“连接向导”包括了“代码生成向导”和“反向工程向导”，它可以用来轻松地实现 Visual FoxPro 类库和 Microsoft Visual Modeler 模型之间的转换。

另外，在 Visual FoxPro 6.0 中，还提供了“数据库向导”，用户可以选择在典型数据库框架基础上建立自己的数据库。关于“数据库向导”的使用将在以后章节中介绍。

1.2.3 OLE 拖放 (Drag and Drop) 技术

在 Visual FoxPro 的早期版本中，也支持表单和控件之间的 OLE 拖放技术。但是，这些拖放技术只适用于 Visual FoxPro 的内部环境。也就是说，用户将无法从其他非 Visual FoxPro 应用程序中拖放任何对象。

Visual FoxPro 6.0 支持真正的 OLE 拖放技术，从而与其他 Windows 的应用程序支持的拖放技术标准相一致。现在，用户可以通过这种技术实现 Visual FoxPro 和其他应用程序之间的数据拖放，如 Microsoft Excel, Word 等。

1.2.4 2 000 年时间的支持

Visual FoxPro 6.0 采用了很多新的特性以保证用户应用程序的 2 000 年兼容性，主要的措施有以下两种。

1. 严格的时间和日期时间格式限定

在 Visual FoxPro 6.0 及 5.0 版本中，采用了严格的日期格式。不管日期设置情况怎样，相同的日期或日期时间值总被赋予严格的日期。这种严格的日期格式为：

```
^yyyy-mm-dd[,][hh[:mm[:ss]]][a | p]]
```

其中，脱字符 ‘^’ 表示了严格的日期格式并使日期和日期时间值采用 ymd 格式。可用的日期及日期时间分隔符有正斜杠 ‘/’、连字符 ‘-’、逗号及空格。

Visual FoxPro 6.0 中增添了新的命令 SET STRICTDATE 以增强日期常量和字符串的 2 000 年兼容性。用户也可以通过选择 Visual FoxPro “工具” 菜单的“选项”命令，在其中的“常规”选择页中设置“2 000 年兼容性”属性。如果希望所有的 Visual FoxPro 均保持这样的设置，可以在选项对话框中单击“设置为默认值”按钮。

2. 世纪属性的设置

在 Visual FoxPro 中可以通过 SET CENTURY ON 命令来打开日期或者日期时间型数据的世纪属性，如：

```
? DATE ()           && 系统将输出 08/04/97
SET CENTURY ON       && 打开世纪属性
? DATE ()           && 系统将输出 08/04/1997
```

在 Visual FoxPro 的早期版本中，输入日期或者日期时间数据可以不用添加世纪值，如 {8/4/97} 的输入格式是合法的。在新版本的 Visual FoxPro 中，当 2 000 年兼容性属性被设置时，日期及日期时间数据的输入必须具有世纪值，任何没有包括世纪值的输入将被认为是非法的。例如：{^1997-8-12 10:45:30 PM} 及 {8/12/1997} 的输入格式是可行的，而输入格式为 {8/12/97} 将被认为是非法。

另外，在 Visual FoxPro 中，增添了一些日期及日期时间函数的功能，这些将在后面的章节中详细介绍。

1.2.5 增强的应用程序框架

新版本的 Visual FoxPro 6.0 提供了功能更强的项目管理器，使用户可以非常方便地生成自己的应用程序。

在项目管理器中，通过使用“应用程序生成器”，可以非常方便地向项目文件中添加数据、表单及报表等数据库资源，并且对它们提供基本选项设置。“应用程序生成器”是一个

比“项目管理器”更加集中、更加快捷的项目文件生成工具。用户只须通过几个选择页之间的切换,就可以对项目信息、项目帮助文件进行设置,并实现对项目中数据、表单及报表的添加、修改及删除等编辑操作。可以说,“应用程序生成器”的使用使得项目文件的管理更加清晰、可靠和方便。

Visual FoxPro 6.0 增添了 ProjectHook 类,以优化对项目文件的管理。在 Visual FoxPro 中,一个项目对象的属性对项目信息对话框中的所有信息都是适用的。由于用户希望在项目文件中实现不同事件的交互作用,例如,新建文件或者连编项目,因此,ProjectHook 类的使用可以支持这些事件代码的交互作用。

当用户项目文件中使用了 ProjectHook 类时,项目文件在打开时被激活(其中包括使用“编译”命令打开一个项目文件),在项目关闭时被退出。

1.2.6 Automation 服务程序的改进

Automation 服务程序是一个组件应用程序,提供了其他应用程序可以通过“自动服务”来使用 and 重用的功能。在 Visual FoxPro 中,可以通过创建 Automation 服务程序来显示可重复使用的表单或者将一个复杂例程包进一个简单组件中供其他程序员使用。

Visual FoxPro 6.0 中的 Automation 服务程序支持套件模型线程。Microsoft Transaction Server 利用标记形成线程化套件服务程序,该服务程序提供了更好的线程保护和可测量性。新的套件模型线程支持没有用户界面的内部处理 .DLL Automation 服务程序和可以有用户界面的外部处理 .EXE Automation 服务程序。新增的 SYS (2335) RP__Unattended_Server_Mode 函数可以用来关闭从远程执行、无须用户干涉的外部处理 .EXE Automation 服务程序的模型事件。

Visual FoxPro Automation 服务程序现在更加强壮,当发生异常时,可以使服务平稳地终止。新增的 COMRETURNERROR() 函数,可用来处理 Automation 服务程序中发生的错误,用户根据相应的信息选择是释放 Visual FoxPro Automation 服务程序还是对异常情况进行处理。

Visual FoxPro 5.0 是按值向 COM 对象(例如在 Visual FoxPro 或者 Visual Basic 中创建的 Automation 服务程序)传递数组;在调用一个方法程序后,数组元素保持不变,并且 COM 对象的改动不会传递到客户程序的元素中。这种限制妨碍了 Visual FoxPro 5.0 和 COM 对象之间来回大量数据的传递。Visual FoxPro 6.0 新增的 COMARRAY() 函数可以指定如何向一个 COM 对象传递数组,并按照下列语法使用 COMARRAY() 函数:

```
oComObject.Method(@MyArray)
```

如果省略@符号,只向 COM 对象传递数组的第一个元素,且 COMARRAY() 函数不起作用。

另外,Visual FoxPro 6.0 包含了一些新的编程功能,以利提高开发效率。编程功能包括 Access 和 Assign 方法程序,用于在查询属性值或试图更改属性值时执行代码。新版本的 Visual FoxPro 还提供对多种图形文件的支持,其中包括对 GIF 和 JPEG 图形的支持。在 Visual FoxPro API 库文件 Foxtools.fll 中还添加了许多文件名处理函数以及新的简化编程任务的语言。

Visual FoxPro 6.0 提供了对旧版本 Visual FoxPro 的完全兼容性。它不仅具有旧版本的所有功能,而且其增强的项目管理功能、扩展的 Internet 工具和关于 2000 年兼容性的保证使得 Visual FoxPro 6.0 都将为用户的数据库操作和功能实现带来更多更大的方便。