

北京科海培训中心

Visual FoxPro 高级编程技术丛书



李春葆 编著

Visual FoxPro 6.0

(应用系统设计篇) 高级编程

科学出版社



内 容 简 介

Visual FoxPro 6.0 中文版是 Microsoft 公司推出的 Fox 系列最新一代的面向对象的数据库开发工具。本书通过 5 个已交付用户使用的 MIS 管理系统的设计与实现过程,深入分析了 Visual FoxPro 编程技术与方法,并提供了这 5 个应用程序的源代码,供读者参考。

全书共分 6 章,内容包括:应用程序开发方法、客户管理系统设计、财政监督管理系统设计、高校工资管理系统设计、路桥收费管理系统设计和进销存管理系统设计。

本书实例清晰、实用性强,适用于 Visual FoxPro 系统编程人员及数据库开发人员。

图书在版编目(CIP)数据

Visual FoxPro 6.0 高级编程(应用系统设计篇)/李春葆编著.

—北京: 科学出版社,2001.3

(Visual FoxPro 高级编程技术丛书)

ISBN 7-03-009302-X

I. V... I. 李... III. 关系数据库-数据库管理系统, Visual FoxPro 6.0—程序设计 IV. TP311.132.3

中国版本图书馆 CIP 数据核字(2001)第 11383 号

科 学 出 版 社 出 版

北京东黄城根北街 16 号

邮政编码:100717

北京朝阳科普印刷厂印刷

科学出版社发行 各地新华书店经销

*

2001 年 4 月第 一 版	开本:787×1092 1/16
2001 年 4 月第一次印刷	印张:34.5
印数:1—5000	字数:839 000

定价:49.00 元(含光盘)

前 言

Visual FoxPro 6.0中文版（本丛书中的VFP即指Visual FoxPro 6.0中文版）是Microsoft公司推出的Fox系列最新一代的面向对象的数据库开发工具。它融入了控件导向和事件导向以及先进的主从结构，不仅适合高效地开发数据库系统，也可以开发包括通信和多媒体系统。

本丛书是作者长期实际开发经验的总结，然后经整理而成，较深入地讨论和分析了VFP编程技术与方法。本丛书包括以下三本：

《Visual FoxPro 6.0高级编程（控件篇）》：该书深入地讨论了VFP中常用控件的特性和使用方法。全书共分24章，内容包括：讨论表单和类的设计、表单控件、标签控件、文本框控件、编辑框控件、命令按钮控件、命令组控件、选项组控件、复选框控件、列表框控件、组合框控件、微调控件、图像控件、表格控件、计时器控件、页框控件、线条控件、形状控件、容器控件、OLE控件和表单集控件的属性、数据表基本操作、数据表查询操作、实用工具设计。每个控件都给出了事件和方法程序的设计实例，并提供了源代码清单，还讲述了实用工具的设计过程。附录中列出了VFP中各种控件的属性、事件和方法程序。

《Visual FoxPro 6.0高级编程技术（ActiveX控件、多媒体和Internet应用设计篇）》：该书深入地讨论了VFP的ActiveX控件、多媒体控件和Internet控件的使用技术。全书共分6章。第1章是ActiveX控件及其使用，讨论了图像列表控件、ProgressBar控件、状态条控件、工具条控件、公共对话框控件、SysInfo控件、TreeView控件、ListView控件、滑动器控件、格式文本框控件、日历控件、三维命令按钮控件、三维组按钮控件、三维面板控件和三维框架控件等。第2章是多媒体控件及其应用，讨论了Animation控件、视频剪辑控件、声音控件、多媒体MCI控件和HWND控件的应用方法，并给出了几个播放器的设计例子。第3章是多媒体API函数及其应用，分类讨论多媒体API函数的应用方法，并给出了多媒体API函数应用的几个例子。第4章是数据发布，讨论了使用“Web发布”向导和_dbf2html类进行数据发布的技术。第5章是启动Web浏览器设计，讨论了使用超级链接控件、_hyperlink类库、_urlcombobox类、_webbrowser4类、自动化技术和API函数启动Web浏览器的设计技术。第6章是发送电子邮件设计，讨论了使用MAPISession和MAPIMessages控件以及使用_mailbtn类发送电子邮件的技术。

《Visual FoxPro 6.0高级编程技术（应用系统设计篇）》：该书给出了使用VFP开发应用的5个实例，并提供了这些应用程序的源代码，供读者参考。全书共分6章。第1章是应用程序开发方法，讨论了使用VFP开发应用系统的一般步骤。第2章给出了一个客户管理系统的设计过程。第3章给出了一个财政监督管理系统的设计过程。第4章给出了一个高校工资管理系统的设计过程。第5章给出了一个路桥收费管理系统的设计过程。第6章给出了一个进销存管理系统的设计过程。

本丛书具有以下显著特点：

- 条理清晰，实例丰富
- 实例功能完善，实用性强
- 提供全部源码，供读者参考

由于作者水平所限，书中难免存在缺点和错误，敬请广大读者指教，作者的E-MAIL为licb@public.wh.hb.cn。

作 者
2001年1月

目 录

第1章 应用程序开发方法	1
1.1 应用程序开发过程.....	1
1.1.1 需求分析.....	1
1.1.2 数据库设计.....	2
1.2 数据库设计.....	4
1.2.1 数据库的设计过程.....	4
1.2.2 创建和删除数据库.....	4
1.2.3 使用数据字典.....	5
1.3 系统菜单设计.....	5
1.3.1 创建菜单系统的过程.....	5
1.3.2 规划菜单系统.....	6
1.3.3 实现菜单系统.....	6
1.4 表单设计.....	7
1.4.1 规划表单.....	7
1.4.2 设计表单.....	7
1.5 报表设计.....	12
1.5.1 规划报表.....	12
1.5.2 设计报表.....	12
1.6 编译应用程序.....	13
1.6.1 构造应用程序.....	13
1.6.2 向项目中添加文件.....	14
1.6.3 联编应用程序.....	14
第2章 客户管理系统设计	15
2.1 系统功能.....	15
2.2 数据库设计.....	15
2.3 系统设计.....	16
2.3.1 主文件MAIN1.PRG.....	16
2.3.2 主表单main设计.....	17
2.3.3 pass表单设计.....	67
第3章 财政监督管理系统设计	71
3.1 系统功能.....	71
3.2 数据库设计.....	71

3.3	系统设计	80
3.3.1	主文件MAIN2.PRG.....	80
3.3.2	JCXT表单设计	81
3.3.3	TH表单设计.....	89
3.3.4	DWINPUT1表单设计.....	97
3.3.5	TH1表单设计.....	102
3.3.6	JC表单设计.....	110
3.3.7	DWINPUT2表单设计.....	119
3.3.8	JC1表单设计.....	123
3.3.9	HZ表单设计.....	130
3.3.10	DWINPUT3表单设计.....	152
3.3.11	HZ1表单设计.....	157
3.3.12	HZ2表单设计.....	169
3.3.13	HZ3表单设计.....	182
3.3.14	JCMC表单设计	195
3.3.15	ZB表单设计	198
3.3.16	SETKM表单设计	223
3.3.17	INIT表单设计.....	229
第4章	高校工资管理系统设计.....	240
4.1	系统功能.....	240
4.2	数据库设计	240
4.3	系统设计	241
4.3.1	主文件MAIN3.PRG.....	241
4.3.2	MAIN主表单设计	242
4.3.3	MAINMENU.MPR菜单程序	245
4.3.4	GZ1表单设计.....	245
4.3.5	GZ11表单设计.....	270
4.3.6	GZ12表单设计.....	273
4.3.7	GZ2表单设计.....	277
4.3.8	GZ21表单设计.....	306
4.3.9	GZ3表单设计.....	316
第5章	路桥收费管理系统设计.....	323
5.1	系统功能.....	323
5.2	数据库设计	323
5.3	系统设计	324
5.3.1	主文件MAIN4.PRG.....	325
5.3.2	PASS表单设计	326
5.3.3	MAIN表单设计	329

5.3.4	FIND表单设计.....	370
5.3.5	主文件MAIN5.PRG.....	381
5.3.6	PASS1表单设计.....	381
5.3.7	MAIN1表单设计.....	386
第6章	进销存管理系统设计.....	395
6.1	系统功能.....	395
6.2	数据库设计.....	395
6.3	系统设计.....	398
6.3.1	主文件MAIN6.PRG.....	398
6.3.2	PASS表单设计.....	399
6.3.3	MAIN表单设计.....	403
6.3.4	JH1表单设计.....	407
6.3.5	JH11表单设计.....	429
6.3.6	JH12表单设计.....	435
6.3.7	JH2表单设计.....	446
6.3.8	JH3表单设计.....	461
6.3.9	JH31表单设计.....	474
6.3.10	JH32表单设计.....	496
6.3.11	JH4表单设计.....	503
6.3.12	JH5表单设计.....	519
6.3.13	JH6表单设计.....	530

第 1 章 应用程序开发方法

随着面向对象语言的产生和运用，面向对象的编程方法已经深入到每个程序员的工作之中。在Windows应用程序的开发过程中，关于界面的代码大约占整个源代码的40%左右，而面向对象的方法可以快速地开发出友好的用户界面，从而节省工作时间，提高工作效率。

一般来说，一个应用程序是由两部分组成的，一部分是界面，另一部分是数据处理，特别是数据库操作。Visual FoxPro，即VFP，是一种面向对象的编程语言，它不仅提供了高效的界面开发方法，同时它又是一种关系数据库管理系统，提供了方便快捷的数据库操作功能。所以，VFP是一种优秀的应用程序开发工具。

1.1 应用程序开发过程

在使用VFP开发应用程序时，主要包括需求分析和数据库设计两个阶段。

1.1.1 需求分析

在设计数据库程序之前，需要认真地规划数据库。在规划阶段，应该让用户更多地参与进来。无论多么仔细的规划，在项目实施过程中也需要不断地进行修改，并认真听取最终用户的反馈信息。

在开发之前所做的设计方案往往会对最终结果产生很大的影响。许多问题都应在深入开发之前加以考虑，例如：这个应用程序面向的用户是谁？用户的主要操作系统是什么？处理的数据集多大？是否需要使用后台数据服务器？是单用户还是网络上的多用户使用系统？

在对数据库的分析中，包括三部分的内容：初始调查、可行性分析和需求分析。在这里主要讨论需求分析的目标。

需求分析阶段的目标是确定开发系统的功能和物理需求。需求分析阶段分为11个主要的任务。

(1) 审查可行性研究数据

其目标是审查保存的可行性研究数据。

(2) 实施需求定义座谈

其目标是搜集数据。该数据是对可行性研究期间搜集数据的扩充。

(3) 定义面向对象的系统需求

其目标是定义由系统执行的功能。

(4) 确定对象

其目标是确定应用程序中指定的每个对象。

(5) 定义客户机和服务器需求（如果有网络的话）

其目标是定义客户机 / 服务器环境的各种需求, 这种环境支持该系统的功能需求和体系结构系统。

(6) 定义信息网络服务需求

其目标是定义信息网络服务级需求。

(7) 定义实现需求

其目标是定义在实现系统中必须满足的物理需求。

(8) 鉴定可供选择方案的标准

其目标是鉴定每个实际的设计系统可供选择的方案。

(9) 建立评价可供选择方案的标准

其目标是建立评价和选择从计算机硬件厂商和第三方软件公司购买的软件包的标准。

(10) 构造需求模型

其目标是构造模型, 该模型能从需求定义期间执行任务的结果中推断出各种需求。该模型应考虑面向对象的客户机 / 服务器计算的各种需求。

(11) 提交需求定义文档

其目标是把从需求分析期间执行任务获得的全部数据存档, 以便检索和维护。

1.1.2 数据库设计

数据库设计包括两个方面: 外部设计和内部设计。

外部设计

外部设计是由需求模型向一组面向用户的设计的过渡, 主要包括9个任务。

(1) 审查需求定义的数据

其目标是审查保存的需求定义数据。

(2) 定义子系统

其目标是设计子系统并给每个子系统分配任务和对象。该任务包括:

- 定义接口对象功能
- 定义实体对象功能
- 定义控制对象功能
- 定义功能子系统

(3) 定义客户机 / 服务器功能并给处理器分配任务

其目标是定义客户机 / 服务器功能并给处理器分配任务。该任务分为:

- 确定处理方式 (基于主机、客户机还是协同处理)
- 分配由客户机执行的功能
- 分配由服务器执行的功能

(4) 定义子系统接口

其目标是定义将出现通信的独立系统或模块之间的公共界面。

- (5) 定义系统的各种安全控制
其目标是定义系统的各种控制。
- (6) 定义各种设计约束
其目标是定义各种系统设计的物理约束、部门约束和工作约束。
- (7) 重复需求设计模型
其目标是重复每一种需求和设计过程,以便把分析和设计过程期间获得的理解融入到系统模型中。
- (8) 评价可供选择的各种设计方案
其目标是对可供选择的各种设计方案进行分析、比较和判断。
- (9) 确定初步设计模型
其目标是给出系统的初步设计模型结构。

内容设计

内部设计是建立内部系统实现和系统测试框架。内部系统设计包括8个部分:

- (1) 设计系统结构
其目标是设计系统设计中必须结合在一起的各个技术部分。
- (2) 完成对象设计
其目标是根据实际的实现环境完成对象设计。
- (3) 确定数据结构
其目标是确定数据结构以便给系统提供输入和输出。该任务包括:
 - 确定事务处理格式
 - 确定显示格式
 - 确定报告格式
 - 确定输出格式
- (4) 设计数据库
其目标是设计数据库,数据库是实现系统设计所必需的。该任务包括:
 - 把若干简单的对象转化为数据库设计
 - 把若干组合对象转变为数据库设计
 - 把若干复合对象转变为数据库设计
 - 把若干关联对象转变为数据库设计
 - 把若干混合对象转变为数据库设计
- (5) 确定各种人工控制过程
其目标是确定与系统设计有关的特殊考虑。
- (6) 确定各种系统控制过程
其目标是确定系统的各个控制过程。该任务包括:
 - 确定I/O控制过程

- 确定用户控制过程
 - 确定访问控制过程
 - 确定文件维护过程
 - 确定恢复控制过程
 - 准备过程手册
- (7) 确定实现和测试方案
其目标是定义系统实现和系统测试方案。
- (8) 构造详细的设计模型
其目标是构造详细的系统设计模型。

1.2 数据库设计

数据库是存储数据的地方。在VFP中,数据是以表的形式存放的,称为数据表。数据表可以属于某个数据库,也可以是自由表,除了前者存在数据字典外,这两种类型的表在关系操作上基本相同。这里讨论的数据库设计主要是指一个应用程序中的数据库设计。

1.2.1 数据库的设计过程

设计一个结构合理的数据库,可为日后整理数据库节省时间,并能更快地得到精确结果。为了快捷、高效地创建一个完善的数据库,必须采取合理的设计步骤。

下面是设计数据库的步骤:

- (1) 确定建立数据库的目的。这有助于确定需要VFP保存哪些信息。
- (2) 确定需要的表。在明确了建立数据库的目的后,就可以着手把信息分成各个独立的主题,例如“职工”或“单位”等。每个主题都可以是一个数据表。
- (3) 确定所需的字段。确定在每个表中要保存哪些信息。在表中,每类信息称作一个字段,在表中为一列。例如在职工表中,可以有“职工号”、“姓名”和“性别”等字段。
- (4) 确定关系。分析每个表,确定一个表中的数据与其他表中的数据有何关系。必要时,可在表中加入字段或创建一个新表来说明。
- (5) 改进设计。对设计作进一步分析,查找其中的错误。例如,在创建的表中加入几个示例数据或记录,看能否从表中得到需要的结果。然后根据需要调整设计。

1.2.2 创建和删除数据库

创建数据库的命令如下:

```
create database <数据库名>
```

在建立数据库后,该数据库是打开的,可以使用以下命令建立该数据库的表:

```
create <表>
```

若数据库不是打开的，可以使用以下命令打开数据库：

```
open database <数据库名>
```

将一个自由表添加到数据库中的命令如下：

```
open database <数据库名>  
add table <表>
```

将一个表从数据库中移去的命令如下：

```
open database <数据库名>  
remove table <表>
```

显示一个打开的数据库的所有表的命令如下：

```
display table
```

删除一个数据库的命令如下：

```
delete database <数据库名>
```

1.2.3 使用数据字典

每个VFP数据库都带有一个数据字典，这样在设计 and 修改数据库时拥有更大的灵活性。这个数据字典是数据库特有的，正因为有了数据字典才使数据库表在功能上强于自由表。

使用数据字典可以创建表之间的永久关系，采用的是alter table命令的add foreign key子句。使用数据字典可以设置字段级规则、表的有效性规则，采用的是alter table命令的set check子句。还可以使用数据字典创建存储过程和设置触发器。

1.3 系统菜单设计

对于任何一个Windows应用程序来说，菜单系统都是其中很重要的一部分，用户在查询信息之前，首先看到的便是菜单。如果把菜单设计得很好，那么只要根据菜单的组织形式和内容，用户就可以很好地理解应用程序的功能。

VFP提供的“菜单设计器”可用来创建菜单，提高应用程序的质量。我们也可以使用编程等方式设计菜单。

1.3.1 创建菜单系统的过程

创建一个完整的菜单系统包括若干步骤：设计菜单系统；创建菜单及子菜单；为系统指定任务，以实现某一具体的功能；生成菜单程序以及运行菜单程序，对菜单系统进行测试。因此，菜单开发过程如下：

规划—>创建—>分配任务—>生成—>测试

创建一个菜单系统，需要如下步骤：

(1) 规划系统

确定需要哪些菜单，这些菜单出现在界面的何处以及哪些菜单要有子菜单等等。

(2) 创建菜单和子菜单

使用菜单设计器或编程方式定义菜单标题、菜单项和子菜单。

(3) 为菜单系统指定任务

指定菜单所要执行的任务，例如移动记录或输入记录等。从菜单设计的角度看，这些任务分为调用下级的“子菜单”、执行的“命令”或执行的“过程”等。

(4) 预览菜单系统

选择菜单设计器中的“预览”按钮或者其他方式预览整个菜单系统。

(5) 生成菜单程序

在菜单设计器中，从“菜单”中选择“生成”即生成相应的.mpr文件。

(6) 运行菜单文件

可以使用do命令直接运行该程序。

1.3.2 规划菜单系统

应用程序的实用性在一定程度上取决于菜单系统的质量。花费一定时间规划菜单，有助于用户接受这些菜单。

在设计菜单系统时，要考虑以下准则：

- 按照用户所要执行的任务组织系统，而不应按照应用程序的层次组织系统。只要用户查看菜单和菜单项，就应该使之对应用程序的组织方法有一个感性认识。因此，要设计好这些菜单和菜单项，必须清楚用户思考问题的方法和完成任务的方法。
- 给每个菜单定义一个有意义的菜单标题。
- 按照估计的菜单项使用频率或逻辑顺序组织菜单项。
- 在菜单项的逻辑组之间放置分隔线。
- 将菜单上的菜单项的个数限制在一个屏幕内。如果菜单项的个数超过一个屏幕，则应为其中的一些菜单项创建子菜单。
- 为菜单和菜单项设置访问键或键盘快捷键。
- 使用能够准确描述菜单项的文字。

1.3.3 实现菜单系统

在VFP中实现菜单系统的方法较多。常用的方法有以下几种：

- 使用菜单设计器。这是最常用的方法。一般对于复杂的系统，菜单层次较多时采用这种实现方法。
- 使用命令按钮。这种方法也经常使用，而且命令按钮具有更大的灵活性，例如可以方便地设计提示文字的字体、颜色等。
- 使用页框。这种方法实现的菜单系统，用户使用非常直观，但这种方法一般只适合

于较简单的系统，因为它不能设计子菜单。

1.4 表单设计

表单是VFP中最常见的界面。利用表单，可让用户在熟悉的界面下查看数据或将数据输入数据库。

1.4.1 规划表单

VFP提供了一个功能强大的表单设计器，使得设计表单的工作变得又快又容易，因此，在设计之前规划表单就显得尤为重要。

在表单设计器中可以处理下列内容：

- 与表单相关联的数据
- 能一起操作的多个表单
- 表单中不同类型的对象
- 顶层表单或子表单
- 基于自定义模板的表单

表单和表单集都是具有属性、事件和方法的对象。在表单设计器中可设置这些属性、事件和方法。表单集包括了一个或多个表单，可将这些表单作为一个整体来操作。

在规划表单时，需要考虑以下几个原则：

- 从用户的角度设计表单。使用用户熟悉的术语，允许用户控制应用程序，按照用户通常完成任务的方式设计表单及表单上的控件。
- 使用户界面尽可能简单和常见。例如，在设计命令按钮时，建立单击动作事件过程而不是双击动作事件过程。
- 要保证界面的一致性。例如，对于输入数据的不同表单应保证它们上面的控件布局相似。
- 不要有太多的限制。允许用户做一些错误的操作，也就是表单应该具有一定的容错性。

1.4.2 设计表单

表单和表单集既然都是具有属性、事件和方法的对象，可以用面向对象的设计方法来处理这些对象。每个表单集是由一个或者多个表单组成的。在VFP中，主要使用表单设计器创建新的表单。

1. 数据环境

每个表单集或者表单都包含有一个数据环境。数据环境是包含表、视图以及表之间关联的对象。在VFP中，可以使用数据环境设计器来设计数据环境，并将其与表单集或表单一起存储。引入数据环境的目的在于：

- 打开或者运行表单时自动打开表和视图(可以通过数据环境中的相应属性取消自动打开和自动关闭功能)。
- 可以通过数据环境中的所有字段来设置控件的ControlSource属性。
- 关闭或者释放表单时自动关闭表和视图。

在“数据环境设计器”中右击，在出现的快捷菜单中选择“属性”，则出现“属性”窗口，在其中可以设置的3个共同的数据环境属性是：

- **AutoCloseTables**: 用于控制在关闭或者释放表单集或表单时，是否自动关闭表和视图。缺省值为真(T.)，表示自动关闭。
- **AutoOpenTables**: 用于控制在运行表单时是否打开数据环境中的表和视图。缺省值为真(T.)，表示自动打开。
- **InitialSelectedAlias**: 用于指定运行表单时所选择的表或者视图。缺省值在设计时是一个空串。如果没有指定，则运行时将把第一个临时表添加到数据环境中作为初始的选择。数据环境是由表、视图以及表之间的关联组成的，因此，必须将表和视图添加到数据环境中，对于添加到数据环境中的表、视图以及关联，都可以从数据环境中移去。而且从数据环境中移去一个表时，该表所涉及的全部关联也将移去。

可以将数据环境设计器中的字段、表和视图等拖动到表单中，即可以通过数据环境来建立表单中的对象。例如，要建立表格控件，则可以将表拖动到表单中；要建立复选框，则可以拖动逻辑型字段；要建立编辑框，则可以拖动备注型字段；要建立OLE绑定型控件，则可以拖动通用型字段；要建立文本框，则可以拖动其他的字段。

通常，在激活“数据环境设计器”后，属性窗口将显示与该数据环境相关的对象和属性等，而且数据环境中的每个表和视图以及表之间的每个关联和数据环境自身都是属性窗口的对象框中的独立对象。

表之间的关联是数据环境中的一项重要内容。对于数据库中具有永久关联的表，当将其加入到数据环境设计器中时，这些永久关联也会自动加入到数据环境中；而对于没有永久关联的表，则可以在数据环境设计器中建立关联。要在数据环境设计器中设置表之间的关联，可以将某一字段从主表中拖到关联表的匹配索引标记上。如果没有索引标记来响应主表中的字段，则系统将提醒我们创建一个索引标记。一旦在数据环境设计器中设置一个关联后，就会在表之间用一条线来指示表之间的关联关系。要改变与关联有关的属性，只需从“属性”窗口的对象框中选择要处理的关联即可。

2. 预定义常量

在表单集或表单中，可以使用一些预定义的常量，因此必须在表单集或表单中包含一个头文件。典型头文件中主要包含用#DEFINE等预处理指示符定义的常量。要在表单集或表单中包含头文件，步骤如下：

- (1) 从“表单”中选择“包含文件”命令。
- (2) 在“包含文件”对话框中指定要包含的文件，或者单击三点“...”按钮打开“包含”对话框后再选择头文件。

(3) 单击“确定”按钮。

3. 表单类型

在VFP中，可以创建以下两种类型的应用程序：

- 多文档界面(MDI)：MDI应用程序由一个主窗口和包含在主窗口中的应用程序窗口组成。VFP就是一个MDI应用程序。
- 单文档界面(SDI)：SDI应用程序由一个或者多个独立的窗口组成，每个窗口单独显示在Windows桌面上。

为了支持创建这两种类型的应用程序，VFP允许创建以下类型的表单：

- 子表单：包含在另一窗口中的表单，可以在创建MDI应用程序时使用。子表单不能移出父表单之外，最小化时显示在父表单的底部。如果父表单最小化，则子表单一起最小化。
- 浮动表单：属于父表单的表单，但不包含在父表单中。浮动表单可以在屏幕上随意移动。如果最小化，浮动表单出现在桌面的底部。如果父表单最小化，则浮动表单也一起最小化。
- 顶层表单：独立的表单，用于创建SDI应用程序，或者在MDI应用程序中作为其他表单的父表单。顶层表单没有父表单，顶层表单出现在Windows任务栏中。

4. 创建表单集

在VFP中，可以将多个表单包含在一个表单集中，这样可以对表单集中的所有表单进行统一操作，例如一起显示或隐藏表单集中的表单，同时将多个表单组织成一个表单集可以统一控制和安排这些表单的相对位置。此外，通过在表单集中建立数据环境，还可以自动同步多个表单中的记录指针，即当我们改变某个表单中父表记录指针位置时，放在其他表单中的子表记录指针将自动调整并显示子表中的记录。而且，启动运行表单集时，表单集中的所有表单将一同装入。

要创建新的表单集，请从“表单”菜单中选择“创建表单集”命令。创建表单集后，就可以将表单分别添加到表单集中。要将表单添加到表单集中，请在“表单”菜单中选择“加入新表单”命令。对于已经包含在表单集中的表单，如果要从表单集中将其移去，则可以在表单设计器中选择要移去的表单，然后从“表单”菜单中选择“移除表单”命令。对于只有一个表单的表单集，则可以从“表单”菜单中选择“移除表单集”命令来移去这个表单集。

在VFP中，表单是以表的形式存入文件中的(扩展名为.SCX)。创建一个表单时，.SCX表文件中包含一条表单记录、一条数据环境记录和两条内部使用的记录。如果向表单或者数据环境中添加对象，则每个对象在.SCX表单文件中拥有一条记录。如果创建一个表单集，则表单集和每个表单在.SCX表单文件中都有一条相应的记录。

注意：创建表单集时，没有必要让每个表单都显示。对于要显示的表单，设置其Visible属性为真(.T.)，否则设置为假(.F.)。

5. 添加对象到表单中

每个表单可以看成是由多个对象组成的屏幕界面。要设计出具有一定功能且满足要求的表单, 必须把适当的控件添加到表单中, 并设置表单和控件的属性值, 编写事件代码。

进行表单设计时, 可以使用“表单控件工具栏”添加标准控件到表单中。VFP所提供的标准控件主要有复选框(CheckBox)、组合框(ComboBox)、命令按钮(CommandButton)、编辑框(EditBox)、图象(Image)、标签(Label)、线条(Line)、列表框(ListBox)、OLE绑定型控件(OLE Bound Control)、OLE容器控件(OLE Container Control)、形状(Shape)、微调控件(Spinner)、文本框(TextBox)和计时器(Timer)。

将VFP标准控件加入到表单时, 必须将控件与表、视图、表字段或者视图字段中的数据绑定在一起, 这可以通过设置控件的ControlSource属性或者表格的RecordSource属性来实现。例如, 在表单中添加一个编辑框, 然后设置编辑框的ControlSource属性为表中的字段名, 这样由编辑框来编辑表中的字段。

此外, VFP中还可以通过拖放项目管理器、数据库设计器或者数据环境设计器中的字段或者表到表单中的方法来创建数据绑定型控件。例如, 将表拖入表单中, 结果将生成一个表格, 表中的字段变成表格列, 字段名变成列标。

6. 添加新的属性与方法到表单中

进行表单设计时, 可以往表单中添加新的属性和方法, 让新属性保存与表单有关的值, 在新的方法中存放调用它时执行的一段代码。这些新属性和方法只能局限在这个表单或表单集中。如果拥有一个表单集, 则新属性和方法将局限在该表单集中。如果没有表单集, 则新属性和方法将局限在一个表单中。

为了将一个新属性添加到表单集或表单中, 其操作步骤如下:

- (1) 从“表单”菜单中选择“新建属性”命令。
- (2) 在“新建属性”对话框中键入新属性的名称和描述该属性的一段说明。

新属性添加到表单集或表单后, 就可以像表单集或表单中原有的属性一样引用。此外, 还可以像其他属性一样添加数组属性到表单中, 其操作步骤如下:

- (1) 从“表单”菜单中选择“新建属性”命令。
- (2) 在“新建属性”对话框中键入数组的名称和相应的维数与大小。例如, 在“属性名”框中键入“ArrayProp[8, 2]”, 则可以建立一个8行2列的二维数组属性。

注意: 数组属性在设计时是只读的, 但可以在运行时进行赋值。

要给表单或表单集中添加一个方法, 必须从“表单”菜单中选择“新建方法”命令, 然后在“新建方法”对话框中键入新方法名和有关的说明。对于用户自定义的新方法, 其调用方式与调用基类方法相同, 都用以下格式:

ObjectName.MethodName

其中, ObjectName是包含要调用方法的对象名称, MethodName是要调用的方法名称。