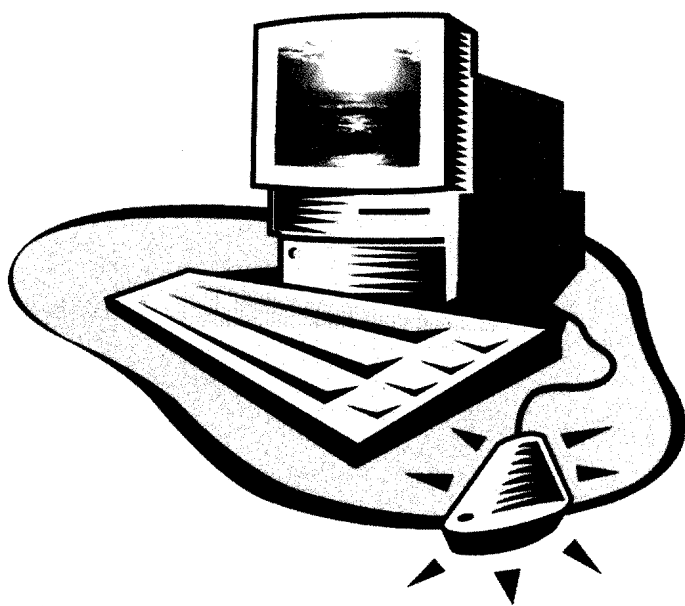


Visual J++

程序设计

魏 泱 刘 烨 笑 然 等编著



- ★ Visual J++ 快速入门
- ★ 建立 Web server 的方法
- ★ 如何创建 Java Applet 程序

北京航空航天大学出版社

内 容 简 介

本书全面系统地介绍了微软最新的 Java 编程工具 Visual J++, 以及如何使用它提供的功能强大的工具快速开发 Java 程序。通过详细的图解和大量的实例介绍了怎样使用 Visual J++ 编写、调试、运行 Java 程序。本书主要内容包括 Visual J++ 快速入门、Visual J++ 的重要工具、建立 Web Server 的方法、创建 Java Applet 程序、开发工具、管理类的工具、AWT 类库的应用、图形处理技术、多线程程序设计、动态效果和动画程序设计、异常处理技术、Java+ActiveX 技术、Cabinet 文件技术、安全技术、应用实例、建立 Java 数据库处理程序、Visual J++ 类库等。

本书适用于计算机专业开发人员、大专院校师生, 以及对 Visual J++ 和 Java 语言感兴趣的读者阅读。

图书在版编目(CIP)数据

Visual J++ 程序设计/魏泱等编. —北京:北京航空航天大学出版社, 1998. 9

ISBN 7-81012-799-3

I. V… II. 魏… III. Java 语言-程序设计 IV. TP312

中国版本图书馆 CIP 数据核字 (98) 第 11340 号

Visual J++ 程序设计

魏 泱 刘 烨 笑 然 等编著

责任编辑 林 红

责任校对 邹坤荣

*

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 62015720)

网址: WWW.buaapress.cn.net

河北涿州新华印刷厂印装 各地书店经销

*

开本: 787×1092 1/16 印张: 24.5 字数: 624 千字

1998 年 9 月第 1 版 1998 年 9 月第 1 次印刷 印数: 4000 册

ISBN 7-81012-799-3/TP·292 定价: 32.50 元

前 言

Java 是一种面向 Internet/Intranet 计算的编程语言。随着“地球村”梦幻般世界的来临,Java 语言及其变体(如 JavaScript)已经广泛渗透到信息高速公路上的各种应用中。对于飞速发展的且已进入到 Internet 时代的计算机技术,掌握 Java 程序设计的软件开发者无疑将成为新一轮技术更新与竞争的宠儿。

Java 是一种与平台无关的编程语言,可以开发在 Internet 上散布的应用程序。它是由 Sun 公司的 JavaSoft 分部开发的一种划时代的新型编程语言。它是一种网络语言,适用于编写互连网络(Internet)这样的广域网上运行的各种应用程序,使原先静止不动的网页在 Java 中因拥有了动画行为而变得灵活、安全且充满生气。

Java 的用途并不仅此。Java 的独特功能使得向数百万的网上用户发送面向对象的应用程序成为可能。随着计算机产业的网络时代的到来,Java 已经宣告了基于网络计算的新纪元的到来。与其它任何一种语言不同,Java 是唯一适合在互联网上分布执行的语言。人们可以从万维网 WWW 上的服务器动态下载 Java 程序,并且既可以在支持 Java 的浏览器上执行也可以单独执行。

Java 作为面向 Internet 的编程语言,一推出就倍受青睐。但是,作为一种编程语言,需要有功能强大的集成开发环境,才能快速、方便地开发出应用程序。Microsoft Visual J++ 正是这样的集成开发环境之一,它使用与 Visual C++ 完全相同的环境 Microsoft Developer Studio,使得许多用户学习起来比较方便。

本书深入介绍了用 Visual J++ 编写 Java 程序的各个方面,内容详实、可读性强。书中的许多内容对于初学者来说都会是耳目一新的,希望能认真阅读,最好能配合上机操作。

由于 Java 语言已经成为一种事实上的程序设计标准,现有的应用程序绝大部分将来都会要求用 Java 语言重新编写,因此计算机专业开发人员都应学习掌握 Java 语言。由于 Java 语言的优秀特性,不夸张地说,Java 语言将成为计算机专业开发人员的必修课,希望读者能早日掌握这门新技术。

本书第一章到第二章由魏泱、徐洁和刘松海编写,第三章到第五章由刘烨、吴青、朱志坚、李宁和顾铁成编写;第六章到第七章由笑然编写,第八章到第九章由张贵军与杨扬编写;第十章和附录由陈群与李敏编写,第十一章到第十五章由翁敬农、魏忠才、朱尽染和郭竟完成。全书由笑然统稿,李梅审校。在此对以上同志一并表示感谢。

编 者

1998.5

目 录

第一章 Visual J++ 快速入门

1.1 Visual J++ 提供的工具介绍	(1)
1.1.1 Visual J++ 的重要工具	(1)
1.1.2 Visual J++ 的其他工具	(2)
1.1.3 Developer Studio 集成开发环境	(5)
1.2 安装 Visual J++	(5)
1.2.1 Visual J++ 的安装环境	(5)
1.2.2 安装 Visual J++	(5)
1.3 运行 Visual J++	(8)
1.4 Visual J++ 的集成开发环境	(10)

第二章 Visual J++ 的重要工具

2.1 Visual J++ 的编译器和解释器	(15)
2.1.1 Java JDK 编译器	(15)
2.1.2 Java 解释器	(16)
2.1.3 Visual J++ 编译器——JVC.EXE	(16)
2.1.4 Visual J++ 解释器——JVIEW.EXE	(20)
2.2 Visual J++ 使用的浏览器	(21)
2.3 Visual J++ 的调试器	(21)
2.3.1 准备调试	(22)
2.3.2 使用调试器	(24)
2.3.3 调试窗口	(25)
2.3.4 掌握调试对话框的操作方法	(30)
2.3.5 调试命令	(34)
2.3.6 程序调试举例	(36)
2.4 Visual J++ 的文本编辑器	(40)
2.4.1 语法分色的概念	(41)
2.4.2 语法分色的设置	(41)
2.4.3 模拟两种流行的编辑器	(42)
2.4.4 两种文本搜索方法	(43)
2.4.5 设置文本编辑器的行为	(47)
2.4.6 设置字体的风格、大小和颜色	(49)
2.5 Visual J++ 的 Applet Wizard 向导	(50)
2.5.1 建立 Applet 程序	(50)
2.5.2 运行向导生成的 Applet 程序	(54)

2.5.3 在 Web 上运行 Applet 程序	(57)
---------------------------------	------

第三章 建立 Web Server

3.1 Internet Information Server 程序介绍	(58)
3.2 建立自己的 Web 站点	(58)
3.3 使用自己的目录	(60)
3.4 设置域名服务	(61)

第四章 创建 Java Applet 程序

4.1 Java Applet 程序的特点分析	(63)
4.2 应用 Applet Wizard 创建 Applet 程序的实例	(65)
4.2.1 创建 Applet 程序	(66)
4.2.2 分析 MyApp.java 源程序	(67)
4.2.3 分析 MyApp.html 文档	(73)
4.2.4 编译 Applet 程序	(73)
4.2.5 建立 HTML 页面	(74)
4.2.6 运行 Applet 程序	(75)
4.3 创建 Java 应用程序	(75)
4.3.1 Java 应用程序的生命周期	(75)
4.3.2 创建 Java 应用程序	(76)
4.3.3 MyApp1.java 源文件一览	(76)
4.3.4 MyApp1Frame.java 源文件一览	(81)
4.3.5 联编 Java 应用程序	(82)
4.3.6 运行项目 MyApp1	(82)
4.3.7 重要说明	(83)

第五章 Visual J++ 的其他开发工具

5.1 Visual J++ 的项目工作区	(86)
5.1.1 项目工作区窗口	(86)
5.1.2 项目工作区	(87)
5.1.3 在 Project Settings 对话框中进行项目设置	(97)
5.1.4 设置目录	(104)
5.2 Visual J++ 的资源编辑器	(105)
5.2.1 资源模板	(106)
5.2.2 使用对话框编辑器	(108)
5.2.3 使用图形编辑器	(117)
5.2.4 调整图像颜色	(125)
5.2.5 使用菜单编辑器	(126)
5.3 Java Resource Wizard 工具	(129)

5.3.1	用 Resource Wizard 进行构件布局	(129)
5.3.2	用 Resource Wizard 实现菜单布局	(142)
5.4	自动化和宏	(145)
5.4.1	实现自动化的方法	(145)
5.4.2	VBScript 宏	(146)
5.4.3	安装运行 VBScript 宏	(150)
5.4.4	录制 VBScript 宏	(152)
5.4.5	Developer Studio 对象	(154)
5.4.6	编写 VBScript 宏	(162)
5.5	定制 Developer Studio	(162)
5.5.1	定制工具条	(162)
5.5.2	定制菜单和菜单项的方法	(169)
5.5.3	Button Appearance 对话框	(171)
5.5.4	定制菜单 Tools	(173)
5.5.5	定制键盘快捷键	(176)

第六章 Visual J++ 中管理类的工具

6.1	ClassView 和 WizardBar 的区别	(177)
6.2	ClassView 功能介绍	(178)
6.2.1	ClassView 的特性	(178)
6.2.2	ClassView 的组成	(178)
6.2.3	用 ClassView 浏览源代码	(178)
6.2.4	ClassView 显示的排序方法	(179)
6.2.5	隐藏和显示 ClassView 视区	(179)
6.3	用 ClassView 为程序增加类、方法或变量	(179)
6.3.1	给项目 ClassDemo 增加新类	(180)
6.3.2	给类 CtrlButton 增加变量	(181)
6.3.3	给类 CtrlButton 增加方法	(181)
6.3.4	实现完整的项目 ClassDemo	(183)
6.3.5	项目的编译执行	(184)
6.4	使用 WizardBar 工具	(185)
6.4.1	显示或隐藏 WizardBar 工具条	(185)
6.4.2	WizardBar 的基本组成	(185)
6.4.3	WizardBar 的跟踪功能	(187)
6.4.4	通过键盘访问 WizardBar	(187)
6.4.5	直接向任意的 WizardBar 控件中输入	(188)

第七章 AWT 类库的应用

7.1	AWT 类库概述	(189)
-----	----------------	-------

7.2 使用 AWT 类库设计用户界面	(191)
7.2.1 构件的布局	(191)
7.2.2 Java 语言的布局管理类	(193)
7.2.3 Java 程序的菜单设计方法	(200)
7.3 再论 Visual J++ 的构件类	(204)
7.3.1 Label 类	(204)
7.3.2 Button 类	(205)
7.3.3 List 类	(207)
7.3.4 TextComponent 类、TextArea 类和 TextField 类	(210)
7.3.5 Scrollbar 类	(213)
7.3.6 Canvas 类	(216)
7.3.7 Checkbox 类和 CheckboxGroup 类	(217)
7.3.8 Choice 类	(219)
7.4 Visual J++ 常用的容器类	(220)
7.4.1 Frame 类	(221)
7.4.2 Panel 类	(223)
7.4.3 Dialog 类	(226)
7.4.4 FileDialog 类	(227)

第八章 Visual J++ 的图形处理技术

8.1 图形坐标系统简介	(230)
8.1.1 Graphics 类的对象	(230)
8.1.2 Graphics 类的坐标系统	(230)
8.2 管理颜色和字体的类	(231)
8.2.1 Color 类	(231)
8.2.2 Font 类和 FontMetric 类	(233)
8.3 常用的绘图方法	(237)
8.3.1 输出字符	(237)
8.3.2 画线	(238)
8.3.3 画矩形	(239)
8.3.4 画椭圆形	(241)
8.3.5 画圆弧	(241)
8.3.6 画多边形	(242)
8.3.7 复制图形	(243)
8.4 载入并显示图形	(244)
8.4.1 载入图形	(244)
8.4.2 显示载入的图形	(246)
8.4.3 载入和显示图形的实例	(246)

第九章 Visual J++ 的多线程程序设计

9.1 多线程简介	(249)
9.2 分析一段多线程的实例	(250)
9.3 创建多线程的两种方法	(251)
9.3.1 方法一——使用继承类 Thread	(251)
9.3.2 方法二——实现接口 Runnable	(254)
9.4 多线程应用中的资源共享	(256)
9.4.1 实例分析	(256)
9.4.2 通过锁定资源协调资源	(258)

第十章 编写实现动态效果和动画的 Applet 程序

10.1 实现动态效果	(260)
10.1.1 实现静态字幕	(260)
10.1.2 使字符串移动	(261)
10.1.3 实现动态效果	(263)
10.1.4 解决字幕移动时的闪烁问题	(266)
10.2 实现动画	(268)
10.2.1 载入与播放图片	(268)
10.2.2 分析一个动画实例	(269)
10.2.3 使用 MediaTracker 类检测图片的载入	(270)
10.2.4 给动画配音	(271)
10.3 根本解决闪烁现象	(272)

第十一章 Visual J++ 的异常处理技术

11.1 了解异常	(275)
11.1.1 异常的定义	(275)
11.1.2 何时出现异常	(276)
11.1.3 检测异常	(276)
11.1.4 异常的层次结构	(276)
11.2 Java 定义的标准异常类	(277)
11.3 异常的处理技术	(278)
11.4 创建一个自己的异常	(280)

第十二章 Visual J++ 的 Java+ActiveX 技术

12.1 通过 VBScript 语言访问 Java Applet 程序	(281)
12.1.1 一个可被 VBScript 语言访问的 Applet 程序	(281)
12.1.2 编写控制 Applet 程序的 HTML 文档	(283)
12.1.3 运行 Applet 程序	(284)

12.2 使用 ActiveX 构件	(285)
12.2.1 什么是 ActiveX 构件	(285)
12.2.2 使用 ActiveX 构件前的准备工作	(286)
12.2.3 使用 ActiveX 构件	(288)
12.2.4 使用 ActiveX 构件的安全性问题	(289)
12.2.5 一个 ActiveX 构件应用实例	(290)
12.3 用 Java 语言创建 ActiveX 构件	(294)
12.3.1 一个简单的 Java 程序	(294)
12.3.2 使用 ActiveX Wizard for Java	(294)
12.3.3 使用新构件.....	(297)

第十三章 Cabinet 文件技术

13.1 Cabinet 文件的概念	(298)
13.2 创建 Cabinet 文件.....	(299)
13.2.1 使用 Diamond 实用程序	(299)
13.2.2 使用 cabarc 实用程序	(300)
13.3 使用 Cabinet 文件.....	(301)
14.3.1 在 HTML 文件中使用 Cabinet 文件	(301)
13.3.2 恰当使用 Cabinet 文件.....	(302)

第十四章 Visual J++ 的安全技术

14.1 预备知识.....	(303)
14.1.1 产生背景.....	(303)
14.1.2 数字签名技术概述.....	(303)
14.1.3 公用密钥证书.....	(304)
14.1.4 证书授权机构.....	(304)
14.1.5 如何获得授权证书.....	(305)
14.2 实现数字签名.....	(305)
14.2.1 信任码技术概述.....	(305)
14.2.2 使用 MakeCert 程序.....	(305)
14.2.3 使用 Cert2SPC 程序	(306)
14.2.4 使用 SignCode 程序	(306)
14.2.5 使用 PeSigMgr 程序.....	(307)
14.2.6 使用 ChkTrust 程序	(307)
14.3 一个数字签名的实例.....	(307)

第十五章 Visual J++ 的应用实例

15.1 应用实例一.....	(311)
15.2 应用实例二.....	(313)

15.3	应用实例三.....	(315)
15.4	应用实例四.....	(321)
15.5	应用实例五.....	(327)
15.6	应用实例六.....	(329)
15.7	应用实例七.....	(332)
15.8	应用实例八.....	(338)

附录 A 建立 Java 数据库处理程序

A.1	设置 ODBC 数据源	(349)
A.2	使用 Dataform Wizard 建立处理数据库中的表的程序	(351)
A.3	编译 Dataform Wizard 向导生成的程序	(354)

附录 B Visual J++ 类库一览

B.1	Java.lang 类包	(356)
B.2	Java.io 类包	(373)

第一章 Visual J++ 快速入门

Visual J++ 是 Microsoft 的 Java 语言开发平台。像任何优秀的集成开发环境一样, Visual J++ 提供了多个开发工具,如文本编辑器、编译器和调试器。除了具备开发环境的典型特征以外, Visual J++ 还提供了几个有用的工具。例如:Java Resource Wizard 可以用来简化用户的界面设计;Java Type Library Wizard 可以支持在 Java 程序中使用 ActiveX 构件。另外,读者还可以根据自己的爱好定制 Developer Studio 环境,使自己的日常工作自动化。

在 Java 语言产生之初,只有 Sun 公司提供的 Java Developer's Kit(JDK)开发环境,随着 Java 语言迅速地成为大家认可的网络编程语言,众多公司都推出了 Java 语言的集成开发环境,例如:Sun 公司的 Java Workshop, Symantec 公司的 cafe,以及 Microsoft 公司的 Visual J++ 1.0 版,不久又推出了 Visual J++ 1.1 版,这就是本书所要介绍的 Visual J++ 开发工具。

1.1 Visual J++ 提供的工具介绍

与任何一个优秀的集成开发环境一样, Visual J++ 为开发 Java 程序提供了许多工具,使程序员可以高效地编写出 Java 程序。

1.1.1 Visual J++ 的重要工具

Visual J++ 包含的最重要的工具有:

- (1) Compiler (编译器):将 Java 源代码变成标准字节码 (bytecode) 可执行代码。
- (2) Debugger (调试器):查找程序中“臭虫”的有利武器。
- (3) Text editor (文本编辑器):Java 源代码编辑工具,可任意定制。
- (4) Applet Wizard:自动生成 Java 项目的起始框架代码。
- (5) Graphic Development Tool (图形化开发工具):Visual J++ 提供的一系列工具,如 Java Resource Wizard (资源向导),使读者更容易开发 Java 程序。

1. Visual J++ 编译器

Visual J++ 使用 Microsoft 自己的 Java 编译器 jvc.exe 把 java 源代码编译成 Java 字节码 (bytecode)。这个编译器与 Sun 公司的 JDK 提供的 Java 编译器 javac 完全兼容,所以读者可以放心地使用 Visual J++ 开发 Java 程序,而不用担心其可移植性。

这个编译器既可以以命令行的形式运行,又可以在集成环境中运行,而且能够按照发行和调试两种模式进行编译。当运行于调试模式时,编译器生成合适的 Java 程序字节码 (bytecode) 代码,用于在 Visual J++ 调试器下运行。程序编译完成后,可以在 Internet Explorer 浏览器中执行或使用 Java 解释器 (jview.exe) 来执行它。

有关 Visual J++ 编译器的详细介绍请参见本书的 2.1 节。

2. Visual J++ 调试器

Visual J++ 调试器使用单步调试命令,并通过众多的窗口和对话框来完成调试任务。

Visual J++ 调试器还提供了众多的窗口,使读者可以时刻观察到项目中变量和表达式,甚至线程的值或状态。这些窗口包括:

(1) Variable(变量)窗口:显示当前代码段中重要的变量值,如当前行、当前方法等等。

(2) Watch(监视)窗口:调试程序时,程序员指定的变量和表达式在本窗口输出。而且,程序员还能在此窗口中改变变量的值。

(3) Disassembly(反汇编)窗口:显示项目的字节码(bytecode)。上下文相关的弹出菜单可用于经编译的字节码(bytecode)代码和文本编辑器中的源代码行之间进行切换。

(4) Quickwatch(快速查看)对话框:这个对话框提供了与 Watch 窗口相同的功能。但 Quickwatch 也许使用更方便,因为可以通过加亮文本编辑器中相应的代码来将变量加入对话框。

有关 Visual J++ 调试器的详细介绍请参见本书的 2.3 节。

3. Visual J++ 文本编辑器

Microsoft 对从前版本的 Developer Studio 的文本编辑器进行了修改,使其可以识别 Java 语法。编辑器采用了语法分色(Syntax Color),对于不同类型的符号,使用不同颜色的字符显示。例如,Java 关键字(如 true,false 和 null)均以蓝色显示;而一般的变量名、常量等文本均以黑色显示;单行和多行注释以绿色显示。同时文本编辑器还能生成并编辑超文本标记语言(HTML)文件。文本编辑器加亮显示与 Java 有关的 HTML 标记如<applet>或<param>以及标记的属性,如高度、宽度等等。

文本编辑器通过鼠标右键提供了一个上下文敏感的弹出菜单,通过这个菜单可以完成诸如拷贝、剪贴等操作。文本编辑器与 Visual J++ 的其他工具,如编辑器、调试器和项目工作区(Project Workspace)窗口有机地集成为一体。例如,当在输出窗口出现一条编译错误时,用户可以双击该窗口的错误信息,文本编辑器将自动跳到出错的代码行。

有关 Visual J++ 文本编辑器的详细介绍请参见本书的 2.4 节。

4. Visual J++ 的 Applet Wizard 向导

Microsoft 历来都喜欢在自己的产品中提供自动生成程序的工具,例如在 Visual C++ 中,提供的 AppWizard 可以帮助程序员自动生成基于 MFC 类库的 Windows 应用程序。

在 Visual J++ 中,Microsoft 也推出了一个自动生成 Java 程序的工具——Applet Wizard。这个工具通过一系列对话框帮助开发者迅速生成 Java 程序的起始框架代码。Applet Wizard 可生成嵌在 HTML 页面中运行的 Java Applet 程序和能够独立运行的 Java 应用程序,或能够同时以这两种方式运行的程序。

有关 Visual J++ 的 AppWizard 向导的详细介绍请参见本书的 2.5 节。

1.1.2 Visual J++ 的其他工具

除了上面几种工具之外,Visual J++ 还提供了许多开发工具,下面介绍几种比较重要的。

(1) 项目工作区(Project Workspace):项目工作区是 Visual J++ 集成开发环境的重要组成部分。它通过带有三个选项卡的窗口,以目录树的形式显示 Books Online 的内容,以及 Visual J++ 项目当中包含的类和文件。

有关 Visual J++ 的项目和项目工作区的详细介绍请参见本书的 5.1 节。

(2) 资源编辑器(Resource Editor):生成对话框、菜单和图形的工具。

有关 Visual J++ 的资源编译器的详细介绍请参见本书的 5.2 节。

(3) Java Resource Wizard:是使用 Windows 资源文件自动为用户界面生成所需要的标准 Java 源代码的有力工具。

有关 Visual J++ 的 Java Resource Wizard 的详细介绍请参见本书的 5.3 节。

(4) Java Type Library Wizard:创建 ActiveX 控件的 Java 语言接口。

有关 Visual J++ 的 Java Type Library Wizard 的详细介绍请参见本书的 12.2 节。

(5) 自动化和宏(Automation and Macros):利用 VBScript 宏使日常的操作自动化。

有关 Visual J++ 的自动化的实现请参见本书的 5.4 节。

(6) 定制 Developer Studio(Customizing Developer Studio):包括对工具条、菜单和其他使用环境的定制。

有关定制 Developer Studio 的详细介绍请参见本书的 5.5 节。

(7) Books Online:提供功能强大的联机帮助功能。

1. 项目工作区(Project Workspace)

集成开发环境通常都使用项目作为程序开发的单位。项目中可以包含多个文件,但是经过编译链接后只会形成一个可执行文件。Visual J++ 也是使用项目来管理多个文件的,这些文件通常存放在同一个目录下。Visual J++ 项目能够包括各种不同类型的文件,例如 Java 源代码、HTML 文件、声音文件和图象文件等。

在 Visual J++ 中,采用了一种更为灵活的项目系统——项目工作区,项目工作区可以包含多个项目,甚至不同编程语言的项目。相同项目工作区中的项目都存储在该项目工作区所在目录下的子目录中。在实际应用当中,当读者试图打开一个项目时,Visual J++ 会提示读者打开包含该项目的项目工作区。也就是说,在 Visual J++ 中已经用项目工作区作为基本的程序开发单位。

项目中还可以包含活动的文档,例如电子表格和 Word 文档文件,这些文件可以在 Visual J++ 集成开发环境中编辑。

2. 项目工作区窗口(Project Workspace Window)

项目工作区窗口是项目工作区的可视界面,所有与项目有关的重要信息都在项目工作区窗口中显示出来。如项目中的类、类中的方法和属性,这种逻辑上的包含关系以树形结构显示。

项目工作区窗口包括以下子窗口:

(1) ClassView:以树形结构显示一个项目中所含的类、方法和属性。最上层的是项目,接着是类,最下层是方法和属性。不同类型的实体使用不同的图标表示。此外,ClassView 图标还标明了访问控制,如 private,public 等等。双击 ClassView 中的类或方法,可在文本编辑器中显示相应的源代码。

(2) FileView:显示项目中包含的各种类型的文件,这种包含是逻辑上的包含关系。在 FileView 子窗口中双击一个文件,该文件就会被调入文本编辑器中显示。

(3) InfoView:除了可以使用该窗口浏览 Visual J++ 的在线帮助的信息外,还可以使用 InfoViewer 浏览 Microsoft on the Web。当 Web 地址(URL)出现在窗口中时,读者可以点击该地址以显示 Web 页面。

3. 资源编辑器(Resource Editor)

资源编辑器是 Visual J++ 用来创建 Windows 资源的工具。由于资源编辑器属于 Developer Studio 开发环境的一部分,所以资源编辑器可以创建加速键(Accelerator)、位图(Bitmap)、光标(Cursor)、对话框(Dialog)、图标(Icon)、字符表(String Table)、菜单(Menu)、工具条(Toolbar)和版本信息(Version)。但是与 Visual J++ 相关的资源只有对话框、菜单和位图这三种。

通常在 Visual J++ 中我们都创建一个资源模板来包含对话框、菜单和位图资源。

4. Java Resource Wizard

在 Windows 编程时我们通常使用 Windows 资源,而 Java 程序的界面设计必须使用 Java 源代码来完成。这有点像从前我们在 DOS 环境使用 Turbo C 编写图形用户界面的方式。这种方式的缺点是,编写起来非常麻烦而且很不直观。

Visual J++ 利用 Java Resource Wizard 帮助程序员进行用户界面设计的工具。它把资源编辑器生成的对话框、菜单资源转换为 Java 源代码,通过这些源代码程序员就可以方便地创建程序的用户界面了。

5. Java Type Library Wizard

Visual J++ 的 Java 虚拟机支持在 Java Applet 程序中使用 ActiveX 构件。在使用 ActiveX 构件之前必须使用 Java Type Library Wizard,把 ActiveX 构件提供的类和接口转换成 Java 可使用的类和接口。

6. 自动化和宏(Automation and Macros)

在使用 Developer Studio 集成开发环境时,我们经常要做一些重复性的工作,例如,每次打开 Developer Studio 时,都打开固定的项目、项目中的某个文件以及帮助文档等等。这些工作完全可以实现自动化。

自动化可以通过 VBScript 宏和插件(Add-ins)来完成。在 Visual J++ 中我们主要使用 VBScript 宏操作 Developer Studio 及其构件来实现自动化。

7. 定制 Developer Studio

工具条和菜单是用户输入命令的主要途径,为了给用户提供灵活的访问命令的方法,

Visual J++ 允许定制工具条和菜单项。利用这个特性,可以完成如下工作:

- (1) 创建新的工具条或菜单;
- (2) 自定义一个已经存在的工具条或菜单;
- (3) 增加或删除菜单命令或工具条按钮;
- (4) 给工具条增加一个菜单;
- (5) 移去一个菜单命令使其成为一个工具条按钮;
- (6) 移去一个工具条使其成为一个菜单命令;
- (7) 拷贝一个菜单或工具条按钮到多个工具条上,这样当某些工具条被隐藏起来时仍然保持可用。

8. 联机帮助(Book online)

当读者使用 Developer Studio 集成开发环境开发 Java 程序时,肯定经常需要参考有关 Visual J++ 和 Java 语言的帮助信

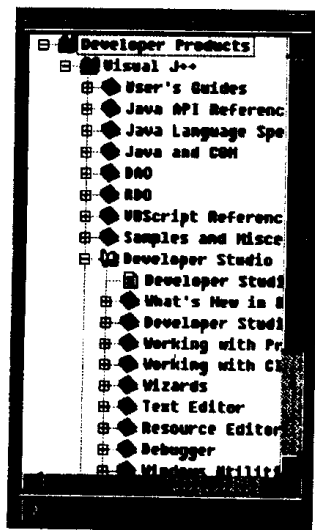


图 1-1 联机帮助的使用

息。这时候读者可以在项目工作区窗口中选择 InfoView 选项卡,通过鼠标点击 InfoView 视图区(pane)中显示的“图书”,读者就可以找到自己需要的信息。如图 1-1 所示。

1.1.3 Developer Studio 集成开发环境

在 Microsoft 的开发工具中 Developer Studio 是一个已经为广大程序员所熟知的工具。Visual J++ 1.0 版也是集成在 Developer Studio 开发环境中的。不过从前的 Developer Studio 集成开发环境,除了 Visual J++ 1.0 之外,集成的是 Visual Test,SourceSafe,Fortran PowerStation 和 Visual C++ 4.0。而新版本的 Developer Studio 除了集成了 Visual J++ 1.1 版外,还集成了下面这些产品:

(1) Visual C++ 5.0 C++ 语言的一个集成开发环境,能快速地开发出功能强大的 C++ 程序。Visual C++ 5.0 的用户界面几乎与 Visual J++ 1.1 的一样。

(2) Visual InterDev 1.0 Internet Web 站点的开发工具包。

(3) Microsoft MSDN 微软的开发网络。

如果以前读者曾经使用过 Developer Studio 集成开发环境,那么再学习 Visual J++ 1.1 开发环境将非常简单。即使读者以前没有用过 Developer Studio 集成开发环境,只要曾经有过使用其他集成开发环境的经验,例如 Turbo C 2.0, Borland C++ 3.1,那么学习起来也不会觉得困难。

1.2 安装 Visual J++

1.2.1 Visual J++ 的安装环境

1. 硬件环境的要求

Visual J++ 1.1 对 PC 机的配置要求为:

- (1) 配有 Intel 486 以上处理器 PC 机(推荐 Pentium 100);
- (2) Microsoft Windows 95 或 Windows NT workstation 4.0 版或更高版本;
- (3) 在 Window 95 中,Visual J++ 至少需要 8MB 内存(推荐 12MB);在 Windows NT workstation 中使用 Visual J++,至少需要 16MB 内存(推荐 20MB 以上);
- (4) 光驱;
- (5) VGA 以上监视器(推荐 SVGA);
- (6) Microsoft 鼠标或兼容设备;

2. Visual J++ 软件包

Visual J++ 软件包上包含下列软件:

- (1) Microsoft Internet Explorer 3.01:Microsoft 的 Web 浏览器;
- (2) Cab&Sign 工具包:压缩 Java Applet 程序(Java 小应用程序)并进行数字签名的工具;
- (3) DaoSDK:数据访问对象开发包;
- (4) DEVSTUDIO:Developer Studio 集成开发环境;
- (5) Vm:Java 虚拟机。

1.2.2 安装 Visual J++

在计算机上插入 Visual J++ 光盘, Windows 95 或 Windows NT 将自动播放该光盘, 启动安装程序如下:

(1) 当 Visual J++ 光盘中的安装程序启动时, 将出现欢迎画面, 如图 1-2 所示如下。

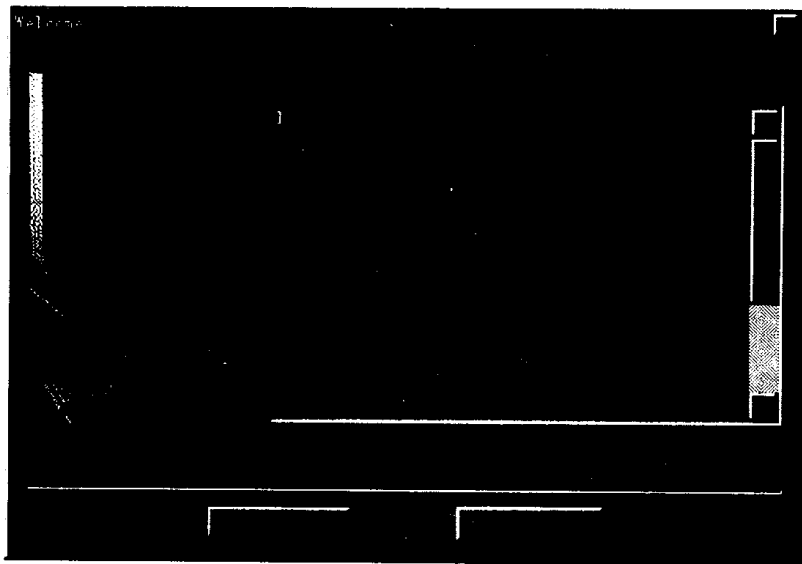


图 1-2 Visual Studio 97 欢迎画面

(2) 选择 Continue 按钮, 将进入注册画面。如图 1-3 所示。

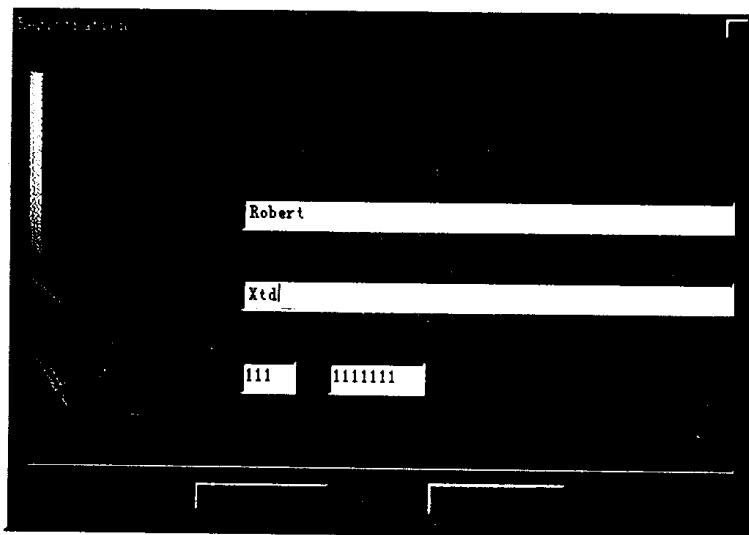


图 1-3 注册画面

(3) 注册画面中的 User 文本框中输入用户名, 在 Organization 文本框中输入单位名, 并输入 CD-Key。然后按下 Continue 按钮, 将进入 Visual Studio 97 主安装画面。如图 1-4 所示。

(4) 在主安装画面中选择 Microsoft Visual J++ 1.1, 将进入 Microsoft Visual J++ 安装欢迎画面, 如图 1-5 所示。

(5) 点击 Next 按钮, 将出现软件许可证协议画面, 如图 1-6 所示。

(6) 点击按钮 Yes, 将弹出安装选项画面, 如图 1-7 所示。

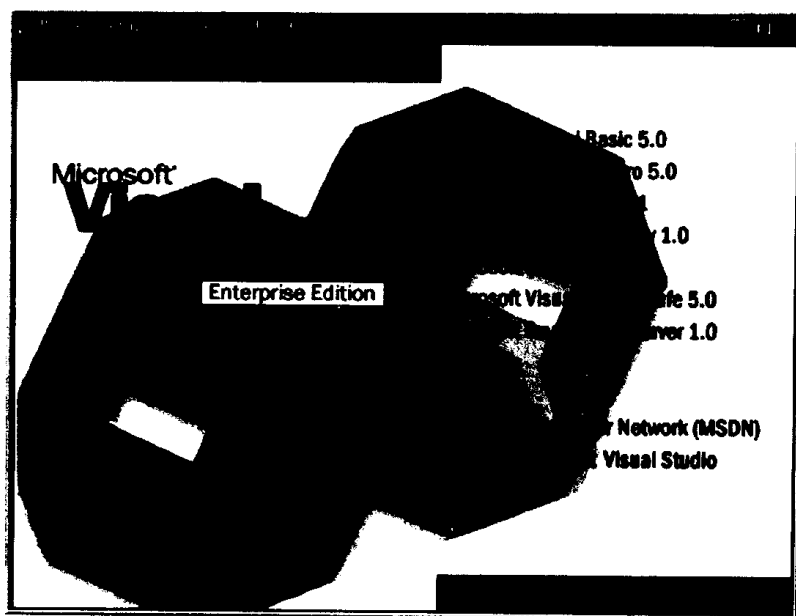


图 1-4 Visual Studio 97 主安装画面

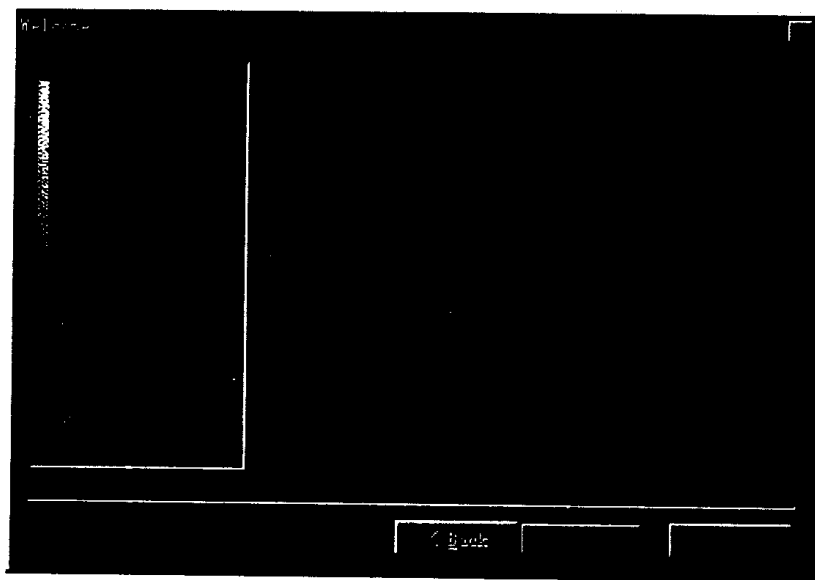


图 1-5 Microsoft Visual J++ 安装画面

(7) 缺省情况下,安装程序默认读者选择典型安装(Typical)。当然读者也可以选择其他选项:最小安装(Minimum)、CD-ROM 及定制安装(Custom Installation)。读者还可以点击 Directory 按钮,改变 Developer Studio 和 Visual J++ 的安装目录。在所有的选择都确定以后,点击 Next 按钮,将出现 Visual J++ 安装画面如图 1-8 所示。

(8) 点击 Next 按钮,Visual J++ 将开始安装。

(9) 安装完成后,将提示读者进行注册。

注意,在安装 Visual J++ 1.1 之前,安装程序将检测当前系统中是否安装了 Microsoft Internet Explorer,如果没有安装或安装了低版本的 Internet Explorer,那么将提示读者安装 Internet Explorer,它存放在读者的光盘上的 VJ1.1 目录中的 IE 子目录中。