

Visual C++ 新起点 ——6.0 实用教程

康博创作室 编著
马 亮 审校

机械工业出版社

Visual C++ 6.0 是 Microsoft 目前综合性最高、最复杂的软件开发产品。本书从最基本的概念入手,以简单易懂的解说,辅以大量的典型应用程序实例及微机运行屏幕显示图例,由浅入深详细系统地介绍了 Visual C++ 6.0 的功能、特点、使用及编程方法和技巧。

图书在版编目 (CIP) 数据

Visual C++ 新起点: 6.0 实用教程/康博创作室编著. - 北京: 机械工业出版社, 1999.3

ISBN 7-111-07006-2

I. V… II. 康… III. C 语言-程序设计-教材 IV. TP312

中国版本图书馆 CIP 数据核字 (1999) 第 00987 号

出 版 人: 马九荣(北京市百万庄大街22号 邮政编码100037)

责任编辑: 于 静

北京市南方印刷厂印刷·新华书店北京发行所发行

1999年3月第1版·1999年6月第2次印刷

787mm×1092mm 1/16·24.75印张

印数: 8 001 - 12 000册

定价: 38.00元

凡购本书, 如有缺页、倒页、脱页, 由本社发行部调换

前 言

Microsoft 公司推出的 C/C++ 在世界软件行业掀起了一场不小的革命，它不但标志着面向对象技术的成熟和完善，使得 C++ 成为举世公认的最优秀的面向对象语言，同时也开创了以面向对象技术为主导的软件设计的新时代，尤其是 Microsoft C/C++ 7.00 所提供的 MFC 库，为人们提供了另外一条开发 Windows 应用的捷径，使人们为之惊叹不已。当世人正沉浸在回味 Microsoft C/C++ 7.00 的绝妙之处时，Microsoft 公司却又不失时机的推出了可视化编程语言 Visual C++，真是一波未平，一波又起。

Visual C++ 6.0 是 Microsoft 目前综合性最高、最复杂的软件开发产品，它提供了前所未有的编程能力和便利性，它所带的多种多样的工具能适合各种编程风格。在 Visual C++ 6.0 中，编辑器、编译器、链接器、调试器、AppWizard、ClassWizard、AppStudio 等工具巧妙而完美地结合在了一起，采用流水线全自动化的编程模式，大大减小了编程的复杂程度，使得编程更加简单、快捷。在这一编程模式下，人们可以利用 AppWizard 来自动生成具有某种特性的程序框架；使用 ClassWizard 往已建立的代码框架中添加所需的类、成员变量及成员函数；利用 AppStudio 来建立应用所需的一系列资源；利用 ClassWizard 来创建相应的成员函数对这些资源进行控制，尤其值得一提的是，Visual C++ 6.0 支持目前广为流行的 Internet 网络编程，相信这也会使广大用户一见倾心。

然而，学习使用 Visual C++ 却不是一件简单的任务，本书首先介绍该软件的一般概念和开发环境，让你选择特定编程任务所需要的具体工具，然后介绍使用这些工具所需的全部信息。本书从最基本的概念入手，以简单易懂的解说，辅以大量的典型应用程序实例及微机运行屏幕显示图例，由浅入深地详细而系统地介绍了 Visual C++ 6.0 的功能、特点、使用及编程方法和技巧。本书内容翔实、循序渐进、图文并茂，给出了大量的学习经验和使用与编程技巧。因此，读者可以花最少的学习时间，以最佳的捷径，从对 C 语言的一知半解，变成能熟练地运用 Visual C++ 6.0 生成视图、文档、分隔窗口、各种控件和对话框，做我们能想象到的各种工作。书中大量的例程都是作者作为软件专业人员精心设计开发的，并经过上机调试与验证。阅读本书之后，你可以充分发挥自己的想象力，开发出各种各样的基于 Windows 的软件产品，临渊羡鱼，不如退而结网，祝愿你利用本书结好大网，在今后的软件开发中大显身手。

本书由康博创作室策划，薛静峰、邹晖和单纯主编，参加本书编写与制作的人员还有曹康、李增民、李娟、许书明、孔祥峰、冯志强、刘利平、刘艳平、王维、刘春晓、王艳春、刘芝泉、吕宏辉、李建萍、周秀会、孙江宏、赵玉双、汪刚、常征、周苏明、王春茫、王强、钟杰、李东升等。由于编著时间仓促，作者水平有限，纰漏之处在所难免，欢迎广大读者多提宝贵意见。

康博创作室

1998 年 12 月

目 录

前言

第 1 章 Visual C++ 6.0 概述	1
1.1 Visual C++ 6.0 的特点	1
1.2 Visual C++ 6.0 的软、硬件环境	2
1.3 Visual C++ 6.0 的安装过程	2
第 2 章 Visual C++ 6.0 的开发环境	9
2.1 工作平台	9
2.2 工具栏	10
2.3 菜单栏	11
2.3.1 “File” 菜单	11
2.3.2 “Edit” 菜单	16
2.3.3 “View” 菜单	19
2.3.4 “Insert” 菜单	21
2.3.5 “Project” 菜单	22
2.3.6 “Build” 菜单	25
2.3.7 “Debug” 工具栏	27
2.3.8 “Tools” 菜单	28
2.3.9 “Windows” 菜单	32
2.3.10 “Help” 菜单	33
2.4 项目工作区	33
2.5 资源	36
2.5.1 资源编辑器	36
2.5.2 资源符号	38
2.5.3 资源符号浏览器	38
2.5.4 对话编辑器	39
2.5.5 菜单编辑器	42
2.5.6 加速键表及其编辑器	45
2.5.7 串表及其编辑器	46
2.5.8 版本信息编辑器	47
2.5.9 图形编辑器	49
2.5.10 工具栏编辑器	52
第 3 章 面向对象编程基础	53
3.1 对象和类的基本特征	53
3.1.1 对象是什么	53
3.1.2 Windows 98 中的对象	53
3.1.3 对象的属性	54

3.1.4 对象的行为	56
3.2 类与对象	56
3.2.1 面向对象的程序设计	56
3.2.2 C++ 中的类类型	57
3.2.3 类和对象的属性及行为	59
3.2.4 如何使用类	61
3.2.5 创建非可视化类	62
3.3 构造函数和析构函数	65
3.4 初始化数据成员	71
3.5 赋值	73
3.6 const 修饰的成员函数	74
3.7 重载	75
3.7.1 类中的函数名重载	76
3.7.2 运算符重载	77
3.8 继承	80
3.8.1 父类和子类	80
3.8.2 访问控制	83
3.8.3 支配规则	89
3.8.4 对构造函数和析构函数的进一步认识	90
第 4 章 C++ 语言简介	92
4.1 概述	92
4.2 什么是类	93
4.3 运算符重载	94
4.4 引用	95
4.5 构造函数	96
4.6 向量	96
4.7 内联扩展	98
4.8 派生类	98
4.9 其他运算符	99
4.10 友元	101
4.11 多态向量	101
4.12 虚函数	102
4.13 C++ 面向对象程序设计	103
4.13.1 对象和类	103
4.13.2 方法和消息	104

4.13.3 继承性	104	FRame 类	142
4.13.4 多态性	105	6.3.2 CWinApp 类	142
4.13.5 动态聚束	106	6.3.3 CWinThread 类	143
4.13.6 多重继承性	106	6.3.4 CCmdTarget 类	143
第 5 章 Windows 编程简介	109	6.3.5 CObject 类	144
5.1 Windows 98 概述	109	6.4 运行时的类型信息	146
5.1.1 新特点	109	6.5 调试支持	147
5.1.2 用户接口	111	6.6 完整的应用程序	150
5.1.3 基层系统体系	117	6.7 消息循环	152
5.1.4 其他方面的改进	117	6.8 MFC 的消息和命令	153
5.2 Windows 98 编程方式	117	6.8.1 消息的种类	153
5.2.1 桌面模型	118	6.8.2 MFC 中的消息处理	153
5.2.2 鼠标	118	6.8.3 消息发送	154
5.2.3 图标和位图	118	6.8.4 用 ClassWizard 进行消息处理	155
5.2.4 工具条、状态条和对话框	118	6.8.5 创建消息映射	156
5.2.5 同应用程序通信	119	6.9 用 MFC 进行绘图和打印	159
5.3 Windows 98 应用程序接口	119	6.9.1 绘制窗口	159
5.4 窗口组件	120	6.9.2 CDC 类	160
5.5 Windows 98 应用程序基础	120	6.9.3 打印	166
5.5.1 WinMain ()	120	6.9.4 打印预览	168
5.5.2 窗口函数	120	6.10 通用类	168
5.5.3 窗口类	121	6.10.1 CString	169
5.5.4 消息循环	121	6.10.2 CTime 类和 CTimeSpan 类	172
5.5.5 Windows 数据类型	121	6.10.3 ColeDateTime 和 ColeDate TimeSpan	174
5.6 Windows 98 程序框架	121	6.10.4 MFC 的文件类	174
5.6.1 定义窗口类	124	6.10.5 CArchive 类	176
5.6.2 创建窗口	126	6.10.6 MFC 的集合类	176
5.6.3 消息循环	127	6.10.7 异常处理	182
5.7 窗口函数	128	6.10.8 MFC 异常类	183
5.8 使用定义文件	128	6.10.9 Win 32 系统级异常	186
5.9 命名协议	129	第 7 章 文档及视图	188
5.10 菜单	130	7.1 文档与视图概述	188
5.10.1 菜单是什么	130	7.1.1 Visual C++ 对文档与视图 的支持	188
5.10.2 建立一个菜单	131	7.1.2 指针变量与引用型变量	189
5.10.3 添加新的菜单列表项	131	7.1.3 探索文档与视图接口	193
5.10.4 建立一个快捷菜单	133	7.2 列表视图控件	194
5.10.5 使用加速键	134	7.2.1 什么是列表视图控件	194
第 6 章 MFC 基础及其通用类	136	7.2.2 列表视图控件的属性	195
6.1 应用程序框架	136	7.2.3 使用一个列表视图控件	196
6.2 使用 AppWizard 创建 MFC 应 用程序	136	7.3 树形视图	199
6.3 几个常用类	142	7.3.1 什么是树形视图控件	200
6.3.1 CDoument, CView 及			

7.3.2 支持 MFC 的树形视图控件	201	8.2.4 启用或禁用按钮	273
7.3.3 用树形视图控件作为一个视图	201	8.2.5 隐藏按钮	274
7.3.4 把树形视图控件加到对话框中	202	8.2.6 定义或设定 Tab 键切换顺序	274
7.3.5 从树形视图中删除条目	204	8.3 编辑控件	275
7.3.6 执行内置的标注编辑	205	8.3.1 理解编辑控件	275
7.4 表单视图	206	8.3.2 编辑控件的属性	276
7.4.1 什么是表单视图	206	8.3.3 将 CEdit 对象与编辑控件相 关联	278
7.4.2 使用表单视图	207	8.3.4 接收从编辑控件中输入的 文本	278
7.5 画笔和画刷	209	8.3.5 用 DDV 和 DDX 例行程序给对 话框传递参数	278
7.5.1 什么是画笔	209	8.4 列表框和组合框	280
7.5.2 用画笔绘画	211	8.4.1 什么是列表框	280
7.5.3 什么是画刷	212	8.4.2 给对话框加入列表框	281
7.6 字体	216	8.4.3 什么是组合框	284
7.6.1 什么是字体	216	8.4.4 使用循环	287
7.6.2 指定字体属性	217	8.5 旋转、进度条、滑块控件	288
7.6.3 用 MFC 来创建字体的例子	220	8.5.1 标准控件	288
7.6.4 选择并配置正确的字体	221	8.5.2 旋钮控件	288
7.6.5 编制字体	222	8.5.3 使用滑块控件	291
7.7 图标	222	8.5.4 使用进度控件	292
7.7.1 什么是图标	222	8.6 位图和图像列表	293
7.7.2 图标的类型	223	8.6.1 什么是位图	293
7.7.3 创建一个新的图标	223	8.6.2 什么是一个图像列表	294
7.8 光标	225	8.6.3 使用图像列表	295
7.8.1 什么是光标	225	8.7 编程举例	297
7.8.2 在 Windows 程序中使用光标	225	8.7.1 创建对话资源	297
7.9 编程举例	227	8.7.2 创建对话类	300
7.9.1 创建应用程序的基本框架	227	8.7.3 定义成员变量	301
7.9.2 视图的处理	228	8.7.4 定义消息处理函数	302
7.9.3 文档的处理	232	8.7.5 修改对话类代码	303
7.9.4 文档数据的串行化处理	236	8.7.6 创建对话类对象	305
7.9.5 视图窗口的滚动处理	237	8.7.7 添加工具栏按钮	309
7.9.6 窗口的分割	238	8.7.8 编译并运行应用程序	311
7.9.7 完整的源程序	239	8.7.9 完整的源程序	312
第 8 章 对话框与控件	258	第 9 章 高级话题	336
8.1 对话框	258	9.1 多任务的实现	336
8.1.1 什么是对话框	258	9.1.1 创建任务	336
8.1.2 使用对话框输入	258	9.1.2 创建多线程程序	339
8.1.3 在 Visual C++ 中创建对话框	259	9.1.3 同步	344
8.1.4 创建基于对话框的项目	266	9.2 动态链接库 DLL	349
8.2 按钮控件	267	9.2.1 使用 DLL 的原因	349
8.2.1 什么是按钮	267		
8.2.2 为按钮设置成员变量	271		
8.2.3 用条件语句调控程序	272		

9.2.2 传统的 DLL	349	9.4.9 一个简单程序	372
9.2.3 MFC 库 DLL	350	9.5 ActiveX	372
9.3 动态数据交换 DDE	355	9.5.1 什么是 ActiveX 控件	372
9.3.1 什么是 DDE	355	9.5.2 为什么要使用 ActiveX 控件	373
9.3.2 利用 DDEML 进行 DDE 编程	355	9.5.3 如何使用 ActiveX 控件	373
9.3.3 DDE 术语	356	9.5.4 一个用 ActiveX 控件定制控件 的例子	376
9.3.4 会晤中的事件序列	358	9.5.5 ActiveX 控件属性、事件和 方法	379
9.3.5 DDEML 函数	365	9.5.6 创建 ActiveX 控件	380
9.4 多媒体应用程序的开发	367	9.5.7 测试 ActiveX 控件	384
9.4.1 数据格式	367	9.6 动态 HTML	385
9.4.2 播放函数	368	9.6.1 什么是动态 HTML	385
9.4.3 声音服务	368	9.6.2 Visual C++ 对动态的 HTML 的支持	385
9.4.4 控制接口	369	9.6.3 使用 MFC 和动态 HTML	387
9.4.5 声音压缩管理器	370		
9.4.6 AVIFile 函数	370		
9.4.7 视频压缩管理器	371		
9.4.8 视频捕获	371		

第 1 章 Visual C++ 6.0 概述

进入 90 年代以来,随着计算机多媒体技术和图形图像技术的蓬勃发展,特别是 Windows 操作系统在微机中的大量使用,可视化技术得到了人们的广泛重视,越来越多的计算机专业人员和非专业人员都开始研究并应用可视化技术。一般来说,可视化技术包含两个方面的含义。一是可视化编程,也就是软件开发阶段的可视化,可视化编程使得编程工作轻松愉快、饶有趣味,给软件开发人员以一种全新的感受。可视化技术的另一含义就是利用计算机图形学的方法和技术,对收集到的大量数据进行处理,并用图形图像的方式直观而具体的显示出来。

要支持可视化编程,就需要有相应的可视化开发环境。而 Visual C++ 就是美国 Microsoft 公司推出的目前使用极为广泛的支持可视化编程的集成环境。

1.1 Visual C++ 6.0 的特点

Visual C++ 6.0 是美国 Microsoft 公司在多年使用过程中不断改进的基础上推出的,用于支持 Win 32、Windows 95/98 和 Windows NT 平台的应用程序(application)、服务(service)和控件(control)开发的工具。

Visual C++ 6.0 在它前一版 Visual C++ 5.0 的基础上做了很大改进,提供了许多新的特点,如下所述:

- 编译器较前一版得到了很大的改进,不但可以支持 ANSI C++ 标准,现在也支持布尔类型,并且对于模板的支持也得到了相当的改善。
- Visual C++ 6.0 开发环境 Developer Studio 是由 Windows 95/98 或 Windows NT 环境下运行的一套集成工具所组成, The Developer Studio 编辑器有了很大的改善,它具有允许编辑器为你自动完成通用语句编辑的特点。使用 Developer Studio,不仅可以创建由 Visual C++ 6.0 使用的源文件和其他文档,而且可以创建、查看和编辑与任何 ActiveX 部件有关的文档。在 Developer Studio 中,可以在项目工作区(project workspace)中组织文件(file)、项目(project)和项目配置(project configuration),可以使用工作区窗口来查看和访问项目中的各种元素。
- 这个开发系统包括一些增加的 MFC (Microsoft Foundation Class) 库的新功能。增加了用于 Internet 编程的库和支持在 Internet Explorer 4.0 和 Windows 98 中编程的新的通用控件。
- Visual C++ 6.0 提供了最快的集成数据库访问,允许用户建立强有力的数据库应用程序,可以使用 ODBC 类和高性能的 32 位 ODBC 驱动程序来访问各种数据库管理系统,也可以使用 DAO (数据访问对象) 类通过编程语言来访问和操纵数据库中的数据并管理数据库、数据库对象与结构。
- Visual C++ 6.0 对 Internet 提供了强有力的支持, Win32 Internet API (WinInet) 使 Internet 成为应用程序的一部分并简化了对 Internet 服务 (FTP、HTTP 和 Gopher) 的访问。

ActiveX 控件可以用在 Internet 和桌面应用程序中, 其文档可以显示在整个 Web 浏览器中, 同时, 可以使用 ChttpServer, ChttpFilter, ChttpServerContext, ChttpFilterContext 和 ChttpStream 类来创建动态连接库以便添加功能到 Internet 服务器和 Web 页中。

- 这个开发环境是和其他安装在计算机中的可视化工具紧密联系的, 例如 Visual J++ 或 Microsoft Developer Network (MSDN) Library。
- 一个增强型的在线帮助系统使 Microsoft Developer Network 变得容易学习, 只需单击鼠标即可实现。若你在机器上安装了 MSDN, 在线帮助系统将自动使用最新的 MSDN 版本。

1.2 Visual C++ 6.0 的软、硬件环境

Visual C++ 6.0 运行所需的软、硬件环境应至少满足以下要求:

- 最新版本的 Windows 或 Windows NT, 例如 Windows 95、Windows 98 或 Windows NT 3.51 或者它们的更新版本。
- IBM PC 机及其兼容机, 最好具有 Pentium 以上的微处理器。
- 至少 20 兆内存 (Windows NT 需要 24 兆内存), 最好使用 64 兆内存或者更多。
- 至少 50 兆自由硬盘空间。
- IE 4.0 或者更高版本的 IE。
- 高密软盘驱动器。
- VGA 显示器。
- CD-ROM 驱动器 (用于联机信息)。

1.3 Visual C++ 6.0 的安装过程

要使用 Visual C++ 6.0 开发应用程序, 首先必须在计算机上安装 Visual C++ 6.0。可以使用安装程序安装新的 Visual C++ 6.0, 也可以将已有低版本的 Visual C++ 进行升级, 如果是升级安装, 安装程序在安装过程中只需复制新的文件或更新过的文件。另外, 安装程序还可以根据实际情况对安装进行削减。

我们使用 Microsoft 公司的 Visual Studio 6.0 软件进行安装, 该软件包括 Visual C++ 6.0、Visual Basic 6.0、Visual FoxPro 6.0 和 Visual InterDev 6.0, 共计两张光盘 A 和 B, 我们只需安装 Visual C++ 6.0 即可。

在 Windows 98 中安装 Visual Studio 6.0 (含 Visual C++ 6.0) 企业版 (Enterprise Edition) 的过程如下:

- 1) 打开计算机启动 Windows 98。
- 2) 在光驱中插入 A 盘。
- 3) 运行 A 盘的 SETUP.EXE, 出现如图 1-1 所示的对话框, 然后用鼠标单击 “Next”。
- 4) 要使用 Visual C++ 6.0, 必须接受一定的协议, 当出现如图 1-2 所示的对话框时, 在单选框中选择 “I accept the agreement”, 然后单击 “Next”。
- 5) 在出现的如图 1-3 所示的对话框中, 键入该软件的 ID 号, 然后输入用户名和所在单位, 最后用鼠标单击 “Next”。
- 6) Visual Studio 6.0 的安装选项有三种 (如图 1-4): Custom、Products 和 Server Applications, 在单选框中选择 “Custom” 之后单击 “Next”。

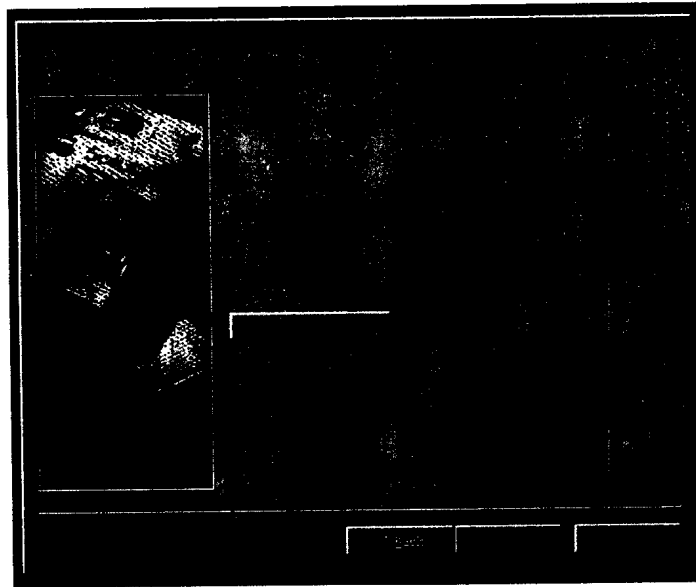


图 1-1 Visual Studio 6.0 企业版安装界面

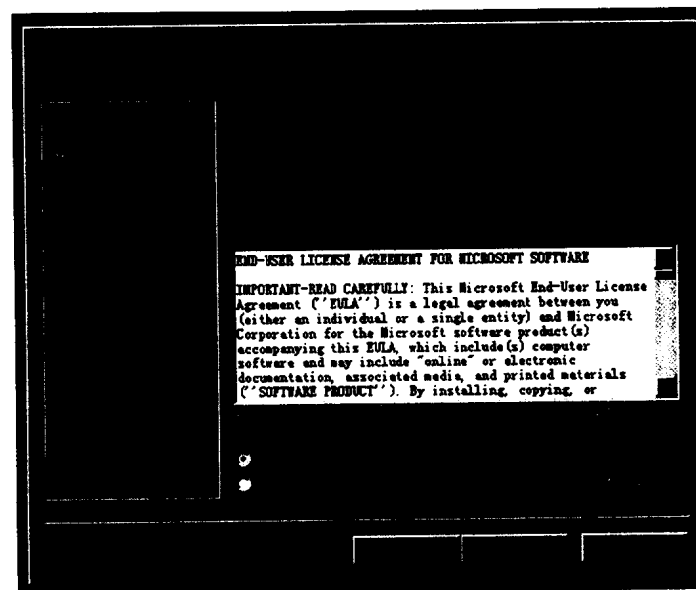


图 1-2 End User License Agreement 对话框

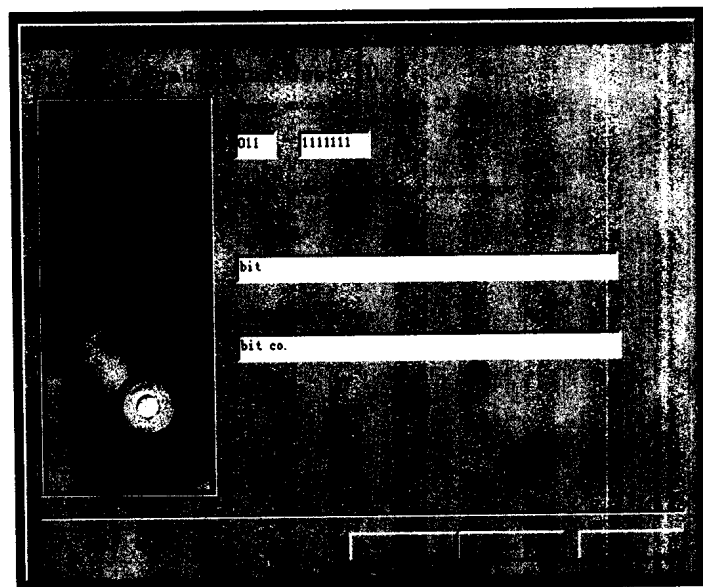


图 1-3 Product Number and User ID 对话框

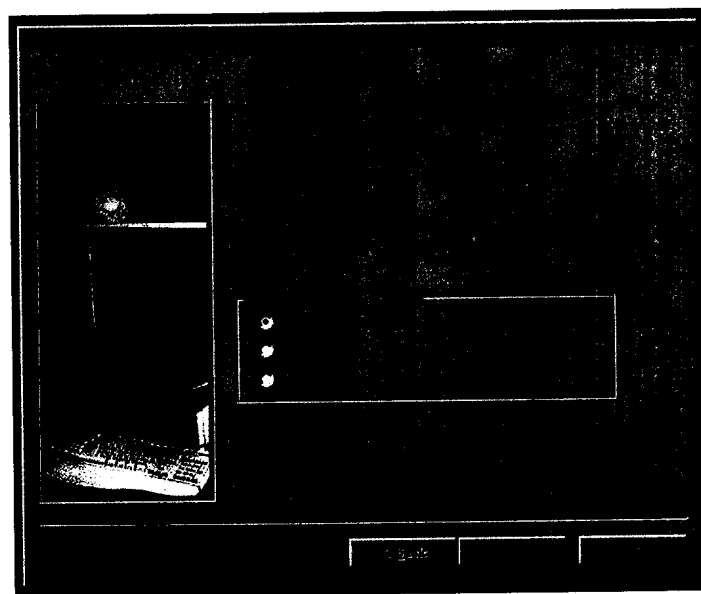


图 1-4 Visual Studio 6.0 Enterprise Edition 对话框

7) 当出现如图 1-5 所示的对话框后, 键入一个路径名, 安装程序将把 Visual C++ 6.0 安装至这个路径下; 缺省时, 将安装在路径“C:\Program Files\Microsoft Visual Studio\”下, 之后再单击“Next”。

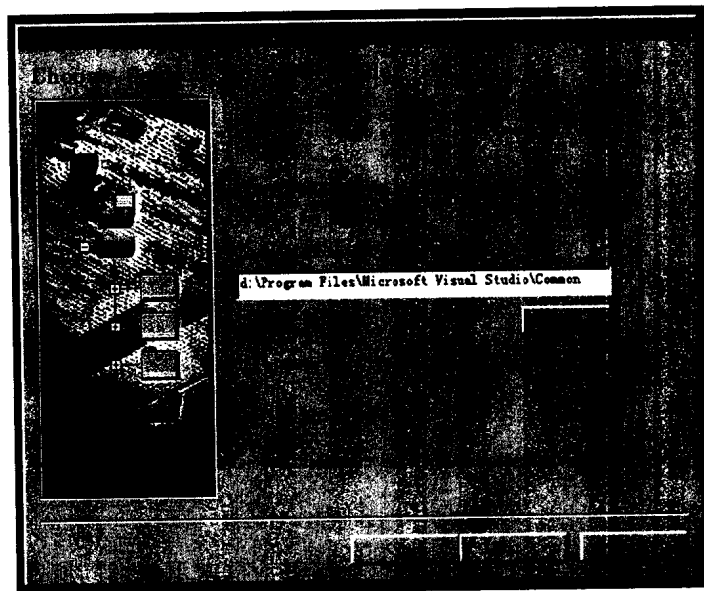


图 1-5 Choose Common Install Folder 对话框

8) 在紧接着出现的如图 1-6 所示的对话框中, 用鼠标单击“Continue”。

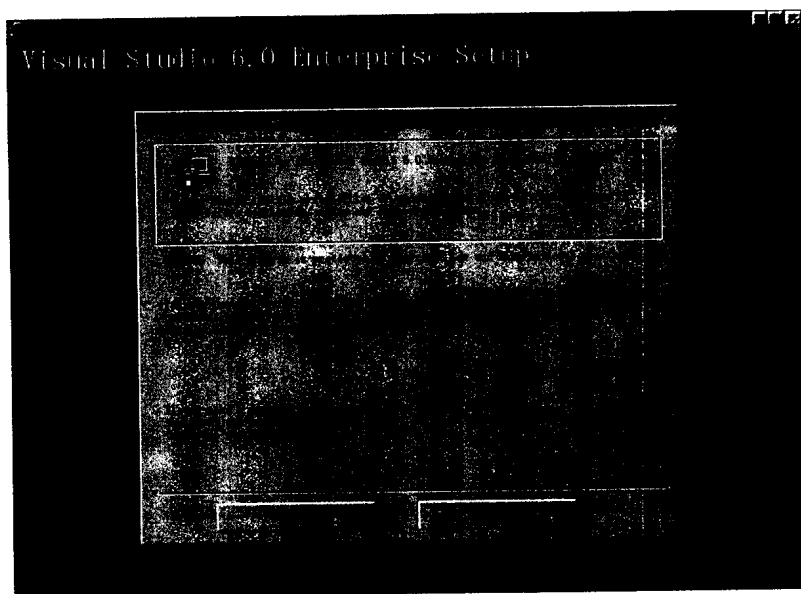


图 1-6 继续对话框

9) 记下如图 1-7 所示的对话框中的产品 ID 号，之后单击“OK”。

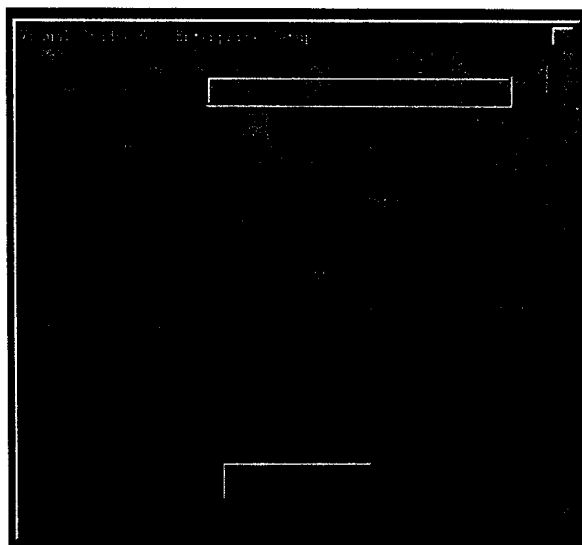


图 1-7 Product ID 对话框

10) 接着就会出现如图 1-8 所示的对话框，该对话框向用户询问所需安装的 Visual Studio 6.0 的内容，我们在单选框中选择所需安装内容，然后单击“Continue”。

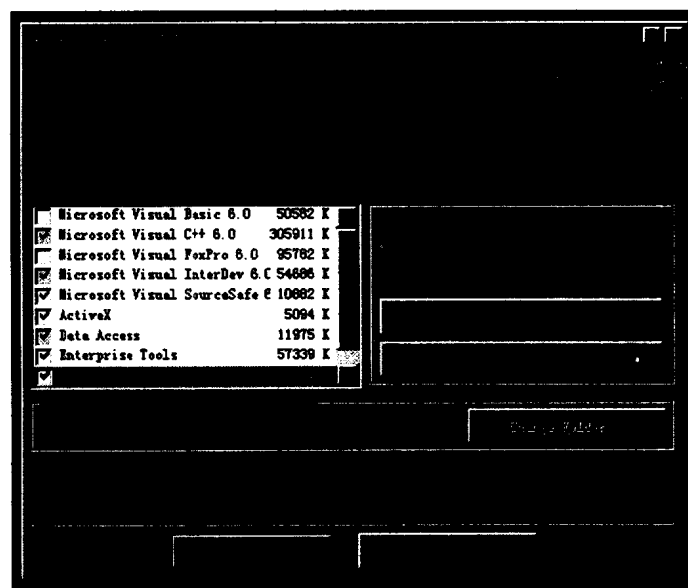


图 1-8 选择安装内容对话框

11) 安装复制过程正式开始, 如图 1-9 所示。

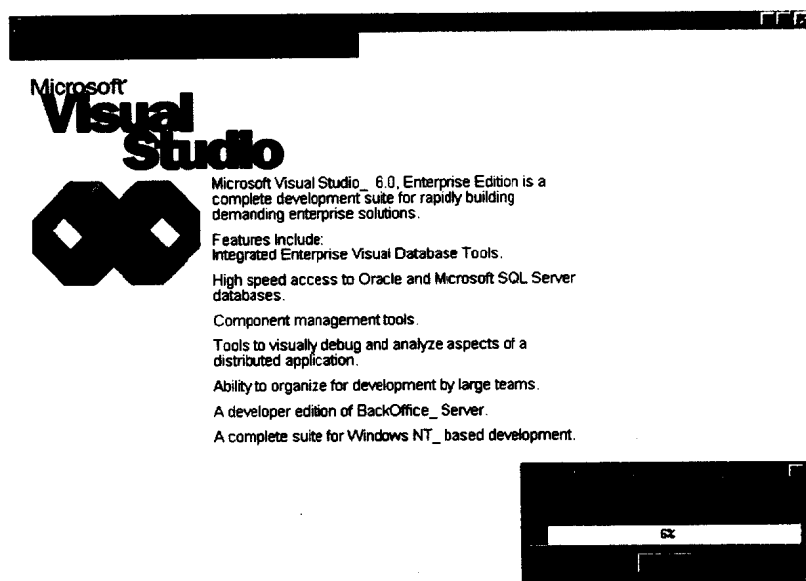


图 1-9 安装复制过程

12) 在复制过程中当提示 (如图 1-10) 找不到某个文件的时候, 换上 B 盘, 单击 “Retry”, 继续安装, 出现又找不到某个文件的时候, 又换上 A 盘, 如此重复, 直至安装过程结束。

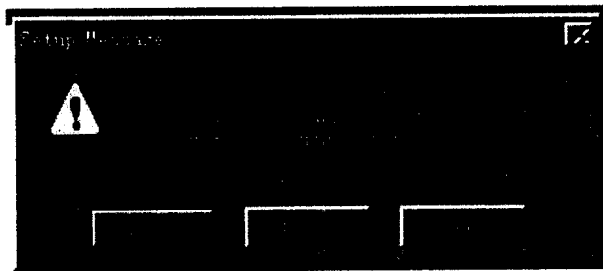


图 1-10 Setup Message 对话框

13) 在接着出现的安装环境变量对话框中单击 “OK” (如图 1-11 所示)。

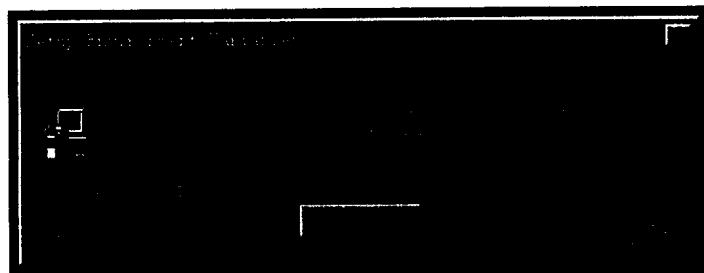


图 1-11 Setup Environment Variables 对话框

14) 下面为了完成安装过程，我们需要重新启动 Windows，在如图 1-12 所示的对话框中单击“Restart Windows”。

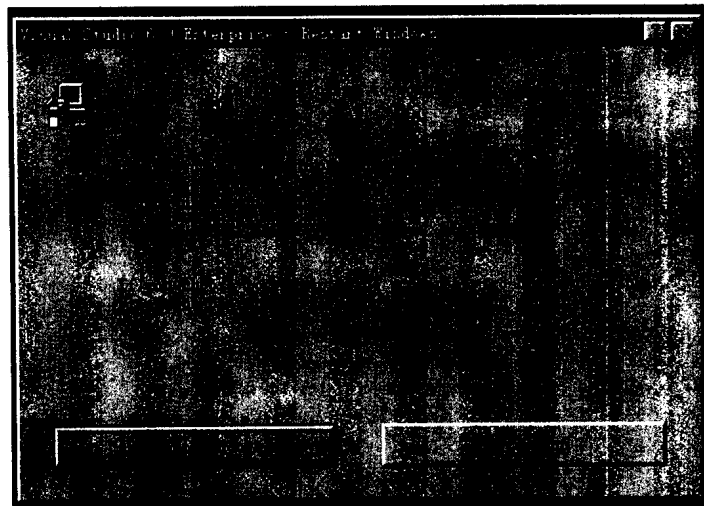


图 1-12 Restart Windows 对话框

至此，Visual C++ 6.0 安装完毕，整个过程大约需要 30 分钟，下面我们就可以学习并使用 Visual C++ 6.0 了。

第2章 Visual C++ 6.0 的开发环境

Visual C++ 6.0 给用户提供了良好的可视化编程环境，该环境是由一套综合的集成工具组成的，在该环境下，可以对 C 和 C++ 应用程序进行各种操作，包括建立、打开、编辑、浏览、保存、编译、链接和调试等，而这些操作都可以用鼠标单击工具按钮来完成。本章就将详细介绍 Visual C++ 6.0 的开发环境，以便让读者了解它的众多功能，利用它熟练地开发各种应用程序。

2.1 工作平台

当我们在 Windows 98 中安装了 Visual C++ 6.0 之后，单击“开始”按钮，从“程序”菜单中启动 Visual C++ 6.0 进入其开发环境（如图 2-1 所示）。

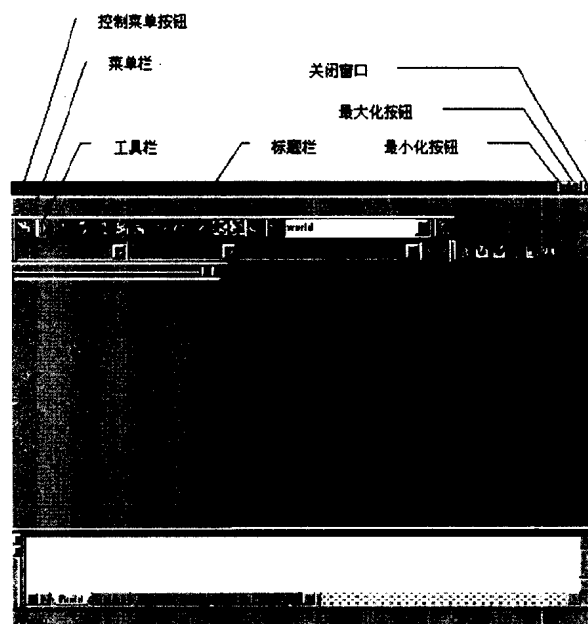


图 2-1 Visual C++ 6.0 开发环境

该开发环境由标题栏、菜单栏、工具栏、工作区窗口、源代码编辑窗口、输出窗口和状态栏组成。

标题栏位于屏幕的最上端，其作用是显示应用程序名和所打开的文件名，标题栏的颜色用于指明对应窗口是否激活。标题栏左边是用于打开窗口控制菜单图标的控制菜单框，用鼠标单击该图标将弹出窗口控制菜单，窗口控制菜单用于控制窗口的大小和位置，如还原、移动、关闭、最大化以及最小化等。标题栏的右边有三个控制按钮，从左至右分别是最小化按钮、还原按钮和关闭按钮，这些按钮用于快速设置窗口大小。例如，使窗口填充整个屏幕、将窗口最小化为图标或关闭窗口。

标题栏的下面是菜单栏和工具栏。工具栏的下面是两个窗口，一个是工作区窗口，另一个是源代码编辑窗口。工作区窗口的下面是输出窗口，用于显示项目建立过程中所产生的错误信息，屏幕最下面是状态栏，给出当前操作或所选择命令的提示信息。

2.2 工具栏

工具栏是由一些操作按钮组成的，分别对应着某些菜单选项或命令的功能。可以直接用鼠标单击这些按钮来完成指定的功能。工具栏按钮使用户的操作过程得到大大简化，并使操作过程可视化。

Visual C++ 6.0 包含有十几种工具栏。缺省时，屏幕工具栏区域显示有两个工具栏，即 Standard 工具栏和 Build 工具栏。

Standard 工具栏主要由以下工具按钮组成。

- New Text File: 建立一个新的文本文件。
- Open: 打开已有的文档。
- Save: 保存文档。
- Save All: 保存所有打开的文件。
- Cut: 把选定的内容剪切到剪贴板中。
- Copy: 把选定的内容复制到剪贴板中。
- Paste: 把剪贴板中的内容插入到当前插入点处。
- Undo: 取消最后一步操作。
- Redo: 重复先前取消的操作。
- Workspace: 显示或者隐藏工作区窗口。
- Output: 显示或者隐藏输出窗口。
- Window List: 管理当前打开的窗口。
- Find in Files: 在多个文件中搜索字符串。
- Find: 激活查找工具。
- Go back: 在被查看的主题列表中回移。
- Go Forward: 在被查看的主题列表中前移。
- Stop Topic Retrieval: 停止主题搜索过程。
- Refresh View: 刷新 InfoViewer 窗口。
- Home Page: 显示 InfoViewer 主页。
- Search: 搜索联机文档。

Build 工具栏主要由以下工具按钮组成：

- Select Active Project: 选择活动项目。
- Select Active Configuration: 选择活动配置。
- Compile: 编译文件。
- Build: 建立项目。
- Stop Build: 停止项目的建立。
- Execute Program: 执行程序。
- Go: 启动或者继续程序的执行。