

Borland C++ 2.0 技术丛书之二(共三册)

Whitewater 资源开发工具

曹恭体 等编译

丛书的内容包括:

- ☐ 《入门》
- ☐ 《用户手册》
- ☐ 《程序设计手册》
- ☐ 《Whitewater 资源开发工具》

北京希望电脑公司

一九九一年八月

前 言

随着计算机程序设计方法的不断提高和改进, 面向对象的程序设计技术在我国也逐渐流行起来, 并有迅猛发展的趋势。它提出了一种全新的程序设计思想, 把数据和对数据的操作融合起来。

美国 Borland 公司在 Turbo C 的基础上, 曾推出过面向对象的程序设计软件包——Turbo C++。它继承并发展了原来 Turbo C 集成环境的优良特性, 并包含了面向对象的基本思想和设计方法, 曾是国际上最受欢迎的面向对象程序设计软件包。

去年五月, 美国 Microsoft 公司推出了 Windows 3.0, 立刻风靡全球, 在微机软件世界引起了一革命。许多软件公司纷纷推出适用于 Windows 3.0 的软件。Borland 公司也不甘落后, 于今年二月推出了适用于开发 Windows 应用程序的 C++ 版本——Borland C++ 2.0。

除了与早期版本的 Turbo C、Turbo C++ 完全兼容外, Borland C++ 2.0 还提供了以下新特色:

- C++: Borland C++ 提供 C++ 编程的全部功能 (AT&T C++ 2.0 版的实现), 并提供了一个 C++ 的类库及对 C++ 1.2 版流的支持。
- ANSI C: Borland C++ 提供至今为止最新的 ANSI C 标准的实现。
- Microsoft Windows 应用程序: 可以用 Borland C++ 编制 Windows 应用程序。Borland C++ 中添加了许多新的功能, 包括资源编译器和资源开发工具。
- 预编译的头文件, 可以缩短程序的编译时间。
- 实模式和保护模式的编译器。
- 类库中提供了集合、数组等类型。
- 求助系统中添加了 Windows API。
- Borland 新的程序员平台: 程序员平台是新型的用户接口, 它胜过了旧的集成环境, 提供对计算机上全范围内程序和工具的访问。包括:
 - ☐ 鼠标器支持
 - ☐ 多重覆盖窗口
 - ☐ 多重文件编辑
 - ☐ 支持内部汇编码
 - ☐ 内部汇编器

☐ 集成调试器

☐ 对大缓冲区的恢复 (undo) 和重复 (redo) 功能

等等。

- VROOMM (面向对象的实时虚拟存储管理): VROOMM使你能很简单地覆盖代码。
- 新的程序员平台的联机访问。
- 联机的快速文本求助, 用拷贝的模拟程序例子试验每一个函数。
- 许多独立的库函数, 包括堆检测函数和完整的复数与BCD数学函数集。

其它特征包括:

- 对-S选择的扩展: C源码作为注释可加到结果汇编代码上。
- 远程对象和大容量数组
- 候补.CFG文件: 用户能创建几个这样的立件, 并且在需要的时刻使用合适的一个。
- 用于命令行编译的响应文件。
- 快速完成复杂的算术运算。

我们在长期从事 Turbo C 和面向对象程序设计的基础上, 根据有关软件和资料编译了这套 Borland C++程序设计丛书。

本丛书共有五册, 分别为《入门》、《用户手册》、《程序设计手册》、《库函数参考手册》、《Whitewater 资源开发工具》, 它们全面系统地介绍了 Borland C++的基础知识和高级技术, 是一套引导读者进行面向对象程序设计和进行 Windows 应用编程的参考书。

在本丛书出版过程中, 得到中国科学院希望高级电脑技术公司资料部秦入华经理、杨淑欣老师的大力帮助和支持, 在此表示衷心的感谢。

编译者

一九九一年五月于北京

目 录

简介.....	(1)
功能.....	(1)
Toolkit 的主要成份.....	(1)
Resource Manager(资源管理器)	(1)
对话框编辑器.....	(2)
位图、光标和图标编辑器	(2)
菜单编辑器.....	(2)
其它编辑器.....	(2)
可处理的文件格式.....	(2)
用户须知.....	(3)
硬件和软件需求.....	(3)
安装.....	(3)
有关本手册.....	(3)
如何入门.....	(4)
第 1 章 初启.....	(5)
在 DOS 下启动	(5)
在 Windows 中启动	(5)
从 Program Manager 中启动	(5)
从 File Manager 或 MS-DOS 运行文件中启动	(5)
添加图标到 Program Manager 中	(6)
退出 Resource Toolkit	(6)
第 2 章 资源和文件.....	(7)
何谓资源.....	(7)
可编辑资源.....	(7)
加速器.....	(7)
位图.....	(8)
光标.....	(8)
图标.....	(8)
对话框.....	(8)
菜单.....	(9)
串.....	(9)
可编辑或存贮的文件.....	(9)
可执行文件	(11)
资源文件	(11)
动态链接库	(11)

资源与对话框描述文件	(11)
位图、光标和图标文件	(12)
头文件	(12)
第3章 资源管理器	(13)
启动编辑器	(14)
创建新资源	(14)
访问现存资源	(14)
选择欲显示的资章种类	(14)
打开现存文件	(15)
编辑资源	(16)
删除资源	(16)
拷贝资源	(16)
文件备份	(17)
创建新文件	(17)
关闭文件	(18)
第4章 加速器编辑器	(19)
创建加速器	(19)
游历编辑器	(19)
编辑正文	(19)
加速表	(19)
Type	(19)
Key	(20)
Code	(20)
Shift	(20)
Ctrl	(20)
Value	(20)
Invert	(21)
Symbol	(21)
头文件	(21)
第5章 位图、光标和图标编辑器	(22)
文件	(22)
位图文件	(22)
OS/2 格式位图	(22)
光标和图标文件	(23)
设置分辨率	(23)
光标和图标编辑器	(23)
位图编辑器	(24)
使用图标编辑器	(24)
使用颜色	(25)

调色板	(25)
为光标和图标图象选择颜色	(26)
Color	(26)
Screen	(26)
Inverse	(26)
调整调色板	(26)
16 色位图	(26)
2 色位图	(27)
保存修改的调色板	(27)
使用用户化的调色板	(27)
绘制和编辑图形	(27)
使用图形工具	(27)
切换反向颜色	(28)
选择线宽	(28)
放大编辑区	(29)
自由线型与阴影	(29)
直线	(29)
受限线	(29)
矩形	(29)
椭圆	(30)
多边形	(30)
滑动工具	(30)
选择图形边界	(30)
Hot Sport 工具	(30)
填充较大的区域	(30)
菜单	(31)
Image 菜单	(31)
Options 菜单	(31)
Tools 菜单	(31)
第 6 章 对话框编辑器	(32)
文件	(32)
头文件	(32)
编辑器概述	(32)
对话框及其控制项	(32)
带菜单的对话框	(33)
创建对话框或控制项	(33)
选择对话框或者控制项	(33)
选择多个控制项	(34)
排除对话框或控制项	(34)

移动对话框或控制项	(34)
修改对话框或控制项的大小	(34)
限制鼠标移动	(35)
设置对话框或控制项的属性	(35)
赋值正文	(35)
设置字模	(35)
修改坐标	(36)
修改控制项的 ID 号	(36)
定义风格	(36)
修改控制项的制表次序	(37)
定义逻辑控制项组	(37)
Tools 控制板	(37)
指针工具	(37)
对话框工具	(37)
坐标对话框	(38)
加标题的对话框	(38)
控制工具	(38)
多风格工具	(38)
压入按钮	(38)
检测框	(39)
单选按钮	(39)
按钮选项的用户化	(39)
编辑控制项	(39)
静态正文	(39)
分组框	(39)
列表框	(40)
Combo 框	(40)
卷滚条	(41)
图标	(42)
用户化的控制项	(42)
对齐控制板	(42)
对齐工具	(42)
菜单	(43)
Dialog 菜单	(43)
Controls 菜单	(44)
Tools 菜单	(45)
Aligment 菜单	(45)
第 7 章 菜单编辑器	(46)
文件	(46)

使用编辑器	(46)
菜单表	(47)
移动按钮选项	(47)
风格和属性域	(47)
测试窗口	(47)
定义菜单正文	(47)
游历编辑器	(47)
编辑正文	(47)
定义菜单层次	(48)
修改菜单条目的位置	(49)
设置激活键	(50)
定义激活键	(50)
在菜单正文中插入制表符	(50)
定义菜单风格和属性	(51)
在弹出式菜单中添加分隔行	(51)
使用检测标志	(51)
定义菜单条目的风格	(51)
对齐栏目的菜单条目	(52)
设置求助属性	(52)
菜单测试	(53)
第 8 章 串编辑器	(54)
定义串	(54)
域和行指示器	(54)
串表	(54)
Symbol 域	(55)
串大小和串值	(55)
游历编辑器	(55)
编辑正文	(55)
头文件	(56)
第 9 章 头文件编辑器	(57)
头文件	(57)
启动编辑器	(57)
创建新的头文件	(58)
打开已存在的头文件	(58)
使用编辑器	(58)
ID 号	(58)
编辑符号	(59)
在头文件编辑器中	(59)
在加速器、串和菜单编辑器中	(59)

在对话框编辑器中	(60)
删除符号	(60)
移动符号	(60)
保存头文件	(60)
第 10 章 公用键及菜单	(61)
表编辑	(61)
表游历	(61)
在编辑域中打入字符	(62)
在选择域中选择选项	(62)
使用菜单	(62)
File 菜单	(63)
New	(63)
Open	(63)
Save	(63)
Save As	(63)
As	(63)
Save Into	(64)
New Header	(65)
Open Header	(65)
Resource Attributes	(65)
Edit 菜单	(65)
Undo(Ctrl+u)	(66)
Cut(Shift+Del)	(66)
Cut Row(Shift+Del)	(66)
Copy(Ctrl+Ins)	(66)
Copy Row(Ctrl+Ins)	(66)
Paste(Shift+Ins)	(66)
Paste Row(Ctrl+Ins)	(66)
Clear(Del)	(66)
Clear Row	(67)
Clear All	(67)
.....	(67)
Select All	
Header 菜单	
Hide	(67)
Show	(67)
附录 A 故障诊断及错误信息	(68)
错误信息	(68)
内存配置	(68)

控制内存分配	(68)
静态和动态内存设置	(68)
检查自由空间	(69)
Windows 和内存	(69)
使用 WIN.INI 控制内存分配	(69)
磁盘空间	(70)
控制板显示	(70)
切换编辑器	(70)

简介

Resource Toolkit(资源开发工具)提供了一些工具,用户可以使用这些工具建造和修改在 Microsoft Windows 3.0 环境下开发的应用程序所用的资源。Resource Toolkit 所提供的工具包括服务于加速器、位图、光标、图标、对话框、菜单和串的编辑器,其中也包括称之为 Resource Manager(资源管理器)的用于资源组织的高效工具。Resource Manager 是资源编辑器的控制中心,除了完成将资源从一个文件拷贝到另一个文件等任务之外,它还具有其它功能。

借助 Resource Toolkit,用户可以在 Windows 环境中建造和管理 Windows 应用程序的用户接口。在资源开发过程中,这种交互性节省了大量时间。此外,Resource Toolkit 可以保存二进制格式的资源,这样就使得资源开发过程不必经历典型的编辑——编译——链接周期。

功能

可以用 Resource Toolkit 创建、编辑和保存的资源有:

- 键盘加速器
- 位图
- 光标
- 图标
- 对话框
- 菜单
- 串

在大多数语言中,用户可以直接使用原代码创建这些资源成份。但是,通过将其定义成独立的资源,就不必使用原代码来管理它们。Windows 能够完成这些管理任务,这样,应用程序的代码可以变得更为流畅,效率也更高。

除了创建资源之外,还可以用 Resource Toolkit 维护 C 和 C++ 格式的头文件以及 Pascal 包含文件。本书第 2 章详细讨论了资源、头文件和包含文件。

Toolkit 的主要成份

可以逻辑地把 Resource Toolkit 划分成两个主要部分,即主窗口部分(Resource Manager)和编辑器部分。

Resource Manager(资源管理器)

Resource Toolkit 的主窗口称为 Resource Manager。该主窗口提供对 Toolkit 中各种特殊编辑器及用户所需其它功能的访问能力。第 3 章讨论了这方面的内容。

对话框编辑器

对话框编辑器(Dialog Box editor)用于创建和编辑对话框。在 Dialog Box 编辑器中创建一个新文件时,它将自动绘制一个加标题的对话框。此外,它支持 Windows 3.0 中新引入的绘图控制功能。二进位组合框结合了编辑控制项与列表框的特点,自画控制指用户化的控制。

位图、光标和图标编辑器

Resource Toolkit 支持新的位图格式窗口,它可用于创建与设备无关的格式。位图、光标和图标编辑器(Bitmap、Cursor、Icon editors)可以读/写这些新的格式,而且完全支持其颜色表。此外,

- 可以用位图编辑器改进调色板的颜色。
- Icon 和 Cursor 编辑器允许用户为每种图标和光标资源创建多幅图象。这多幅图象使得图标和光标与设备无关。当某个应用程序在运行过程中调用一个图标或光标时,Windows 会自动加载与在其 Setup 程序中指定的设备驱动器相适应的图象。
- 可以利用 Bitmap、cursor 和 Icon 编辑器中的 Pouring 工具,以某些色彩填充任意区域。
- Bitmap、Cursor 和 Icon 编辑器中有一个 Pixel Size(象素大小)工具,用户可以以三种大小的象素之一进行绘图:1×1、4×4 或 8×8。

菜单编辑器

菜单编辑器(Menu editor)支持 Windows 3.0 中引入的分层菜单,分层菜单可以让用户为每个菜单条目创建多于一级的弹出式菜单。在 Menu 编辑器中,可以通过缩进 Text 域中的正文来创建弹出的级别或层次,可以在用于定义高层菜单条目正文的同一表格中来定义弹出式菜单条目的正文。这样,创建和编辑弹出式菜单就变得较为方便,修改条目在菜单上的位置以及条目在菜单层次中的级别的过程也变得更为容易。

其它编辑器

其它编辑器包括加速器和串编辑器(Accelerator and String editors)。Accelerator 编辑器可以让用户创建和编辑加速器资源;加速器指用以组织命令的热键。String 编辑器让用户创建和编辑串资源;串资源指应用程序所用的正文串,通常指窗口中的信息或标题。

可处理的文件格式

可以将 Resource Toolkit 用于 Windows 环境中的任何应用开发项目。它支持与 Windows 兼容的任何程序(例如用 Borland C++ 2.0 编写的程序),且可用于维护可执行文件(.EXE)、资源文件(.RES)、动态链接库(DLLS)、位图文件(.BMP)、光标文件(.CUR)和图标文件(.ICO)中的资源。

尽管 Resource Toolkit 是以二进制格式编辑和保存文件的,但它也可以将资源保存到正文文件中,条件是:该正文文件必须与 SDK 用以编译资源的资源描述文件(.RC)兼容。

用户须知

Resource Toolkit 运行于 Windows 3.0 或其后续版本中。欲使用 Resource Toolkit, 必须了解如何运行 Windows, 本手册假定读者已经熟悉 Windows 和一些 Windows 应用程序, 而且手中备有 Windows 系统随带的相关文档。用户至少应知道以下几方面:

- 使用鼠标。
- 移动和缩放窗口; 激活窗口。
- 卷滚窗口。
- 选择菜单、列表框和对话框中的选项或文件。
- 使窗口区域最大; 将窗口恢复到以前的大小。
- 使窗口或应用缩小成图标形式。
- 在 Windows 3.0 或更高版本中使用 Program Manager 工具。

硬件和软件需求

必须加载 Windows。Resource Toolkit 可以在运行 Windows 的大多数计算机上运行。这些计算机包括 IBM PC/AT、PS/2 及其兼容机、基于 386 的计算机(如 Compaq Deskpro)等, 但不限于这些机型。

欲运行 Resource Toolkit, 用户计算机必须有如下配置:

- Windows 3.0 或后续版本。
- 1MB 的 RAM。
- 图形显示适配器(EGA 或性能更佳的适配器)。
- 鼠标或其它定位设备。
- 用以保存临时文件的至少 700k 的磁盘空间。如果打开的 .EXE 文件较大, 那么, 所创建的临时文件可能与 .EXE 文件一样大, 所以要占用更多的磁盘空间。为了优化性能, 建议用户至少预留 3MB 的磁盘空间。

打印机是可选的。Windows 支持鼠标、图形显示适配器和打印机。对于 Windows 所支持的设备, Resource Toolkit 亦能使用之。例如, Resource Toolkit 支持绝大多数 IBM 图形适配器以及其它类型的适配器。Windows 也支持多数鼠标和其它定位设备, 利用 Windows Setup 程序可以显示出所支持设备的完整列表。

安装:

Borland C++ 安装程序将自动安装 Resource Toolkit。

有关本手册

本手册的章节组织形式如下:

第 1 章, “初启”, 告知如何启动和结束 Resource Toolkit 中的某一功能。

第 2 章,“资源和文件”,提供一个资源概述,讨论可在 Resource Toolkit 中使用的文件。它告诉用户如何使用这些文件去开发 Windows 应用程序的资源。

第 3 章,“资源管理器”,告知如何使用 Resource Toolkit 的主窗口,即 Resource Manager。

第 4 章到第 9 章,每章讨论了如何使用一种特定的 Resource Toolkit 编辑器。

第 10 章,“常用键及菜单”,讨论如何使用 Resource Toolkit 游历和编辑 Accelerator、String 和 Menu 编辑器的表格,以及如何使用其选项为各编辑器所公用的菜单:File、Edit 和 Header 菜单。

附录 A,“故障诊断和错误信息”,讨论如何解决在使用 Resource Toolkit 过程中所碰到的问题;告知用户当某些故障意外地中断了编辑过程时如何恢复磁盘空间;提供了一些内存控制信息。

如何入门

为了深入了解 Resource Toolkit,建议用户

- 遵循第 1 章的示教,学习如何启动和退出 Resource Toolkit。
- 继续阅读第 2 章的内容,以概要了解资源并学会如何使用 Resource Toolkit。
- 阅读第 3 章,以了解如何使用 Resource Toolkit 的主窗口,即 Resource Manager。
- 根据用户自己的需要阅读其它章节。

第 1 章 初启

本章介绍如何启动和退出 Resource Toolkit,同时交待了如何将 Resource Toolkit 的图标添加到 Windows Program Manager(窗口程序管理器)中去。

可以在 DOS 提示符下或者 Windows 中启动一种或多种 Resource Toolkit 功能。一旦启动了 Resource Toolkit,Resource Manager 就被打开,且出示版权信息框,欲关闭版权信息框,可以按下 Enter 键或者 OK 鼠标按钮。

在 DOS 下启动

欲在 DOS 提示符下启动 Resource Toolkit:

1. 将目录修改到 Resource Toolkit 的主目录下。如果主目录为 \WRT,则可打入:

CD \WRT

并按下 Enter 键。

2. 当处于 Resource Toolkit 的主目录下时,打入

WIN WRT

并按下 Enter 键,这样就启动了一个 Resource Toolkit 功能。文件 WRT. IMA、WRT. DAT 和 WRT. FON 必须与文件 WRT. EXE 处于同一目录下。

在 Windows 中启动

在 Windows 中,用户可以从 Program Manager、File Manager 和 MS-DOS 运行文件中启动 Resource Toolkit。

从 Program Manager 中启动

欲从 Program Manager 中启动 Resource Toolkit:

1. 启动 Windows,激活 Program Manager。
2. 激活包含 Resource Toolkit 图标的窗口。如果还没有将 Resource Toolkit 图标添加到 Program Manager 中,则可参阅后续内容,以了解如何添加图标,以及如何在没有图标的情况下退出。
3. 在 Resource Toolkit 图标上按两次鼠标按钮。

从 File Manager 或 MS-DOS 运行文件中启动

欲从 File Manager 或者 MS-DOS 运行文件中启动 Resource Toolkit:

1. 打开包含 WRT. EXE 文件的目录。
2. 在文件名上按两下鼠标按钮,这样就启动了一个 Resource Toolkit 功能。文件 WRT. IMA、WRT. DAT 以及 WRT. FON 必须与 WRT. EXE 处于同一目录下。

添加图标到 Program Manager 中

如果已将 Resource Toolkit 图标添加到了 Program Manager 中,则可以在此图标上按两下鼠标按钮,从而启动 Resource Toolkit。

欲将图标添加到 Program Manager 中:

1. 启动 Windows,激活 Program Manager.
2. 激活欲将图标添加于其中的窗口。
3. 从 Program Manager 菜单上选择 File|New,以打开 New Program Object 对话框。
4. 选择 New 域中的 Program Item,然后按下对话框中的 OK 按钮以接受该设置。这样就打开了 Program Item Properties 对话框。
5. 在对话框的 Description 域中打入希望显示在图标之下的名字。例如,用户可能希望在图标在显示 Resource Toolkit。
6. 在对话框的 Command Line 域中打入

[path] WRT.EXE

其中可选项 path 是指包含此文件的目录,例如:

C:\WRT\WRT.EXE

定义 WRT.EXE 文件的目录为 C:\WRT\,

7. 如有必要,在 Change Icon 上按下鼠标按钮,以修改图标。
8. 按下对话框中的 OK 按钮,这样就将图标添加到了活动的窗口中。
9. 欲启动 Resource Toolkit,可在此图标上按两次鼠标按钮。

退出 Resource Toolkit。

欲退出 Resource Toolkit:

1. 在 Resource Manager 的 Control 菜单上按两次鼠标按钮。
2. 或者在 Control 菜单上按下鼠标按钮,再选择此菜单上的 Exit WRT 选项。

第 2 章 资源和文件

本章概要说明可以在 Resource Toolkit 中处理的资源和文件。其中也介绍了如何使用文件开发 Windows 应用程序中的资源。

何谓资源？

窗口应用程序由一序列不同的图形元素所组成。例如：一个应用中可能含有 10 个不同的鼠标和光标，以表达不同的操作模式。可以不将光标信息显式地保存到应用程序的代码中，而作为在需要时可加载到程序中的资源来单独存放。用这种方法分离出资源的好处在于：

- 使得在多种应用程序中使用同种资源的过程变得较为容易。
- 不修改或者不访问应用程序的原代码本身即可修改应用程序的用户可见画面。

可以根据数据类型对资源进行分类——位图、光标、对话框等等——而且可以对每种类型赋予一个独立的名称或编号。数据可以是与资源相关的任何信息。例如，一个对话框资源可能包含表达三种按钮、按钮标号的数据，其中，按钮将被绘制在屏幕上。

资源描述了程序元素的可见属性。它并不定义功能，事实上，修改资源时并不会对功能码产生影响。从原代码中分离出资源时，就可以独立处理源代码和资源。这表明，用户可以使用 Resource Toolkit 修改 Windows 应用程序的可见部分，但无需访问其原代码。

可以用 Resource Toolkit 浏览、编辑、拷贝和删除某些资源。用户能够拷贝、换名和删除但不能编辑或浏览的资源是：

- 字模，应用程序使用它显示正文。
- .RC 数据文件，原始数据资源。
- 用户定义的资源，指需要添到可执行文件中的任何数据。

下一节介绍可以编辑、浏览、拷贝和删除的资源。

可编辑资源

本节描述用户可以用 Resource Toolkit 编辑的资源类型，即：

加速器	对话框
位图	菜单
光标	串
图标	

加速器

加速器指按键或者组合键，它可以代替为了激活某个命令而选择菜单选项的操作。加速器换取了应用程序的执行速度和操作的简易性。例如，相对于下拉出 Edit 菜单并选择 Paste