Windows 3.1和Windows 95系列丛书

Borland C++ Object Windows 程序设计指南

方 旭 张克强 曲文路 等编

北京航空航天大学出版社

Windows

图书在版编目(CIP)数据

Borland C++ ObjectWindows 程序设计指南 / 方旭等编.
—北京:北京航空航天大学出版社,1995.10
ISBN 7-81012-592-3

I.B… Ⅰ.方… Ⅱ.窗口(软件)-程序设计 Ⅳ.TP317

中国版本图书馆 CIP 数据核字(95)第 13533 号

内 容 简 介

本书是针对利用 Borland C++ 4.0、4.5 或更高版本进行 C++ Windows 程序设计的人员编写的,全面系统地介绍了利用 Borland C++ ObjectWindows 进行 Windows 程序设计。主要内容包括:利用 C++和 ObjectWindows 编写 Windows 应用程序原理、程序设计技术(包括 GDI 图形操作、文本处理、菜单、对话框、键盘输入、加速键和鼠标设计)、ObjectWindows 类库的类、解释类的数据成员、函数成员、所在的头文件、消息响应表、事件处理程序、调度函数等。附录给出了从 ObjectWindows 1.0 移植到 ObjectWindows 2.0 的方法。书中的程序设计技术也适用于 Windows 95 和 Windows NT。本书适合于所有使用 C++编写 Windows 应用程序的读者。

●书　　名: Borland C++ Object[illegible]指南
●编 著 者: 方　旭　张克强　曲[illegible] 等编.
●责任编辑: 曾昭奇
●出 版 者: 北京航空航天大学出[illegible]社
●地　　址: 北京学院路 37 号(100[illegible])　[illegible]015720(发行科电话)
●印 刷 者: 通县觅子店印刷厂
●发　　行: 新华书店总店科技发行所
●经　　售: 全国各地新华书店
●开　　本: 787×1092　1/16
●印　　张: 45
●字　　数: 1150千字
●印　　数: 5000 册
●版　　次: 1995 年 10 月第 1 版
●印　　次: 1995 年 10 月第 1 次印刷
●书　　号: ISBN 7-81012-592-3/TP・176
●定　　价: 55.00 元

前　言

ObjectWindows 2.0 是在 Windows 3.1、Win32s 和 Windows NT 中编写 Borland C++ 应用程序的框架，可以让用户快捷而容易地建立充满特色的窗口应用程序。ObjectWindows 2.0 提供了如下特色：

- 16 位和 32 位工作平台间的易移植性
- 自动的消息解包
- 可靠的例外与错误处理
- 由于它没有使用特殊的编译器和语言扩展，允许程序拿到别的编译器和环境下使用
- 包容 Windows GDI 对象
- 易进行数据提取和显示的文档/视图类
- 打印机和打印预浏览类
- 支持 Visual Basic 控制
- 输入验证处理

本书第一章至第十六章以面向任务的方式提出许多论题，用来描述怎样使用 ObjectWindows 类集中的功能组去实现不同的任务。各章的内容如下：

第一章："ObjectWindows 综述"对 ObjectWindows 的层次结构作了一主要而非技术性的浏览。

第二章："学用 ObjectWindows"分 12 个教学步骤来介绍 ObjectWindows 2.0 的一些特征。

第三章："应用程序对象"描述了应用程序对象和应用程序类 TApplication。

第四章："界面对象"讨论了界面对象在 ObjectWindows 2.0 编程模型中的使用。界面对象是指诸如表述窗口、对话框和控制等的类，这些类都基于类 TWindow。

第五章："事件处理"解释了响应表——ObjectWindows 2.0 处理事件的方法。

第六章："窗口对象"描述了包括如何使用框架窗、层面窗、装饰框架窗以及 MDI 窗等窗口对象。

第七章："菜单对象"讨论了菜单对象的用法和 TMenu 类。

第八章："对话框对象"解释如何使用对话框对象(如 TDialog 和 TDialog 得到的对象，同时包括了基于 TCommonDialog 类的窗口通用对话框)。

第九章："文档/视图对象"介绍了用到 TDocument 类、TView 类和 TDocManager 类的 ObjectWindows 2.0 文档/视图编程模型。

第十章："控制对象"讨论了不同控制的使用，例如按钮、列表框、编辑框等等。

第十一章："小工具和小工具窗口对象"解释了包括控制条、状态条、按钮工具等等一些工具和工具窗口。

第十二章："打印机对象"描述了如何使用打印机和打印预浏览类。

第十三章："图形对象"提供了包容 Windows GDI 的类。

第十四章："验证对象"描述了在编辑控制中输入验证生成器的使用。

第十五章："可视基控制对象"讨论了使用 Visual Basic 控制和 TVbxControl 类在 ObjectWindows 设计中的应用。

第十六章："ObjectWindows 动态链接库"解释了包容 ObjectWindows 的动态链接库(DLL)。

第十七章至第二十章是 ObjectWindows 的参考，帮助用户了解 ObjectWindows 中的各个方面：查阅每一类的全面论述；学习如何使用特殊的 ObjectWindows 类及其数据成员和成员函数的详细细节；可以看到 ObjectWindows 类之间直接的或非直接的多种继承关系；学习哪些类可以引入或重新定义函数；确定一个类的父类引入数据成员或成员函数；学习怎样定义数据成员和成员函数；使用事件处理函数响应消息和使用发送函数传送 Windows 消息。

第十七章："库参考信息"以字母表顺序列出所有标准的 ObjectWindows 类，包括它们的说明解释、使用方法及其数据成员的说明。它也阐述了一些类要使用的非对象元素，像结构、常量、变量以及宏等。

第十八章："事件处理程序"列出了 ObjectWindows 的函数及传送 Windows 消息要注意的一些问题。

第十九章："调度函数"列出发送 Windows 消息的所有 ObjectWindows 函数。

第二十章："封装的 WIN API 函数"列出了包容封装 Windows API 函数的对应 ObjectWindows 函数。

附录 A："ObjectWindows 1.0 到 ObjectWindows 2.0 的转换"表述了怎样转换用户的 ObjectWindows 1.0 应用程序才能在 ObjectWindows 2.0 下正常运行。

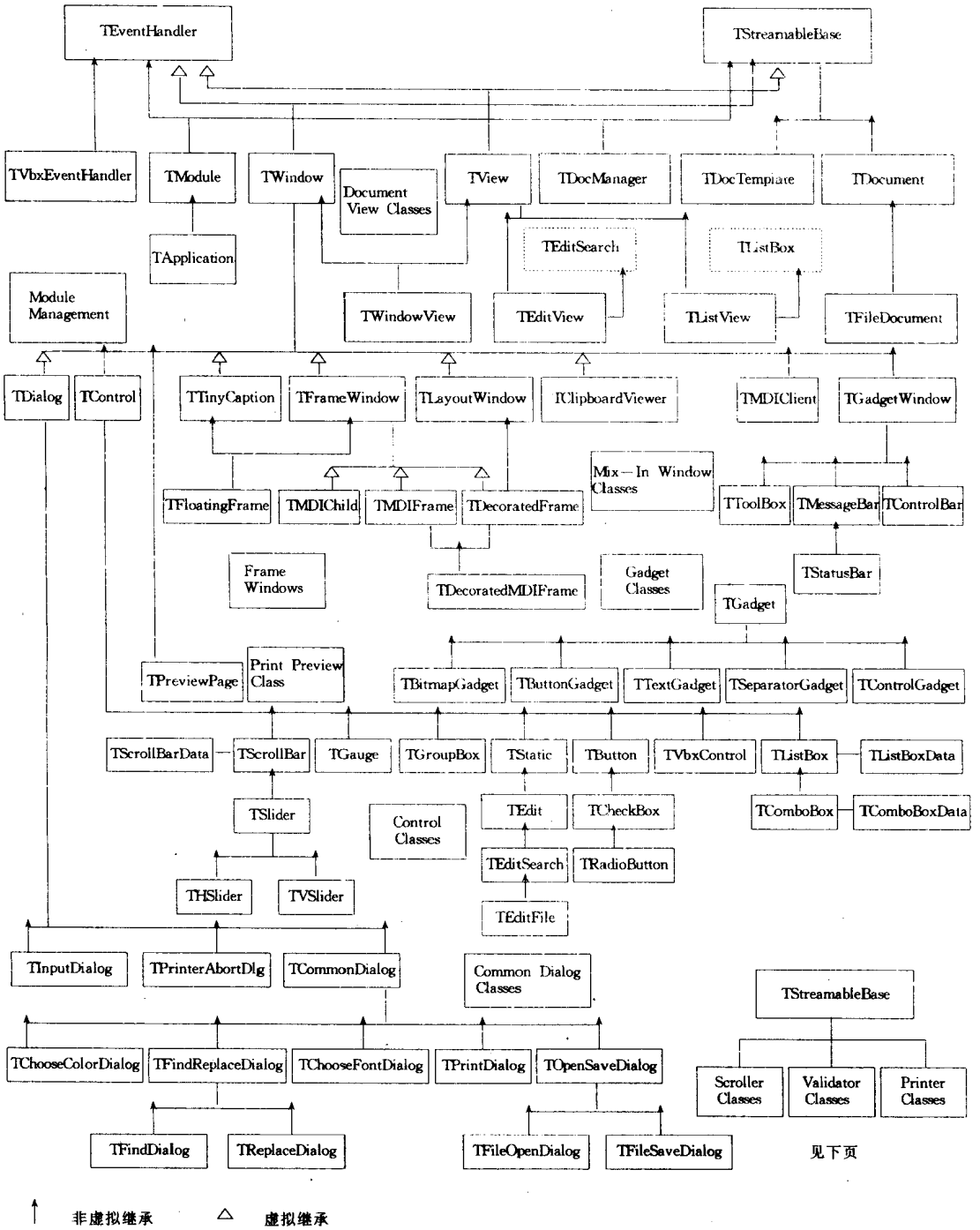
ObjectWindows 的继承图显示出了本书中阐述的各种类。这些类按照功能类别分组，所有相关的类放在一块阴影区域里。某一类如果其父类是多重继承类，就用虚线包括起来。例如，TListBox 是 TListView 的父类，它由 TView 和 TListBox 共同派生。下图就是这些类的结构。

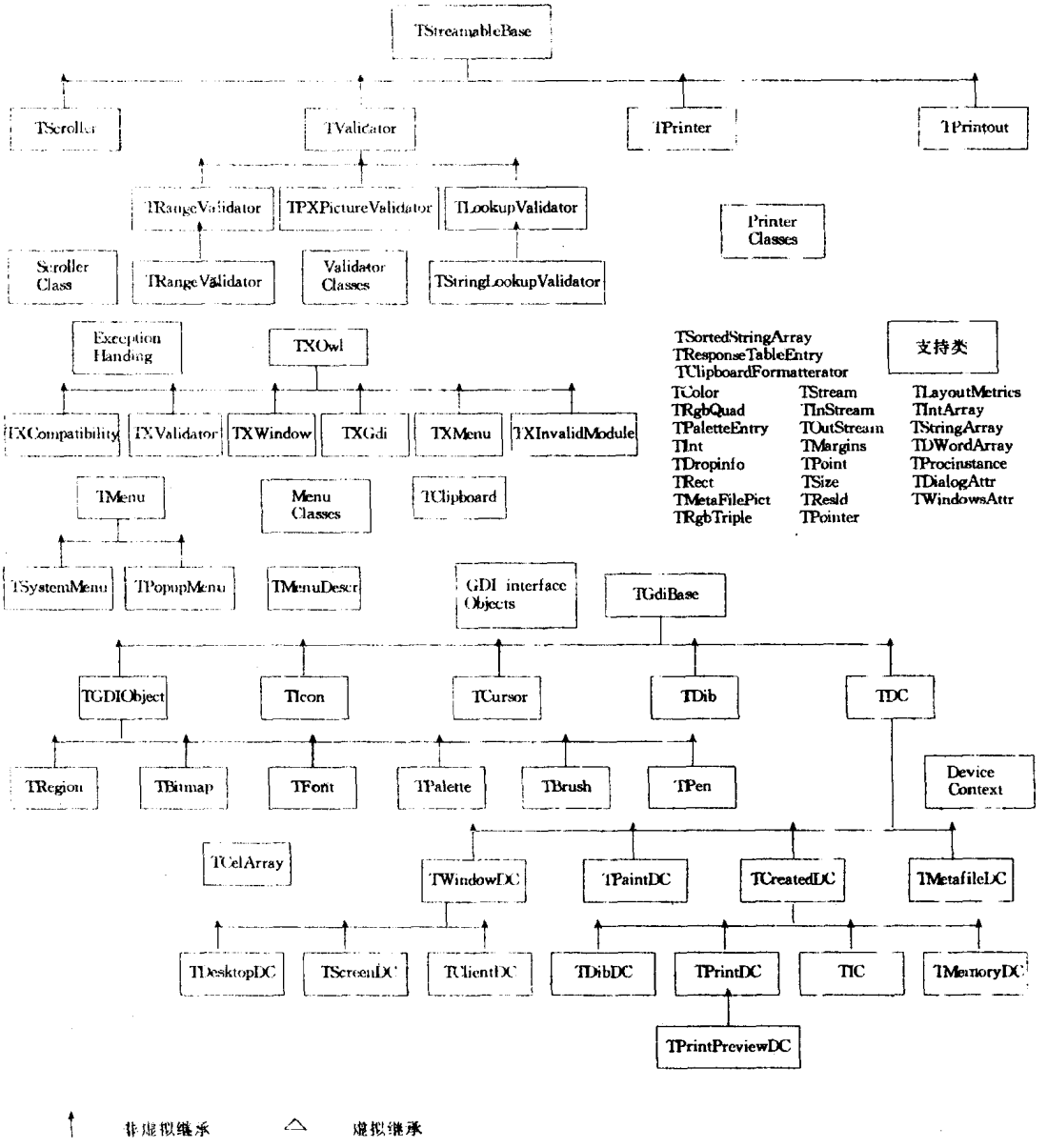
在 Borland C++中包含两个部分，即 C 语言和C++语言。C 语言是C++语言的前身，并符合 ANSI C 标准的 C 语言。如果想转而学习或使用C++语言来编程，同样，Borland C++中的面向对象的品性也是一流的。

所谓的 OWL 是指 ObjectWindows Class Library，即 Borland ObjectWindows 类库，它是 Borland C++中用C++编写 Windows 应用程序的类库。OWL 封装了 Windows API(应用程序编程接口)的函数、数据结构和宏，以面向对象的类提供给C++的程序员。

编写过 Windows 应用程序的读者，都有这样的体会：一个编写好的的 Windows 应用程序是很容易使用的；然而，开发这样的程序并不是一件容易的事。为此许多程序员就不得不熟悉编写 Windows 应用程序所必需的 500 多个 Windows API 函数。

Borland 用来解决上面难题的方法是使用 OWL。可重用的类很容易掌握和使用。OWL 充分利用了C++提供的数据抽象的特性，它的使用简化了 Windows 编程。初级程序员可以像用"烹调全书"炒菜那样"照搬照抄"，有经验的C++程序员可以扩展其中的类或者把它们混合进自己的类层次中。





OWL 库提供用于管理 Windows 对象的类，并提供许多可以同时用于 MS－DOS 和 Windows 的通用类。比如，创建和管理文件、字符串、一致的存储和异常处理都是通用的类。

实际上，ObjectWindows Class Library 虚拟表示了每个 Windows API 的功能，包含了用于简化消息处理、诊断及所有 Windows 应用程序都需要重复处理的一般问题的高级代码。Windows API 函数的逻辑组合和增强的优点主要有以下九个方面：

- Windows API 的封装是完整的。OWL 库为所有的经常使用的 Windows API 性能提供支持，包括窗口函数、消息、控件、菜单、对话框、GDI(图形设置接口)对象(字体、画刷、画笔及位图)、对象链接，以及多文档界面(MDI)。

- OWL 很容易学。Borland 做了最大的努力以保证 OWL 函数和相关的参数的名字与 Windows API 父类相一致。这极大地减少了有经验的 Windows 程序员想利用能简化工作的 OWL 平台的优点时所可能带来的混乱。它也可以帮助初级 Windows 程序员利用 OWL 这个 Windows API 的超集来完成自己的工作。
- C++代码更加有效。当在小内存模式下编译 OWL 库中的类时，一个应用程序只耗用少许的额外 RAM。一个 OWL 应用程序的执行速度几乎同使用标准 Windows API C 语言所编写的程序相等。大部分的 OWL 应用程序的运行速度仅比相应的 Windows API C 语言程序慢 5%——考虑到 OWL 有助于缩短程序设计开发周期，这是非常值得的。
- OWL 库提供了自动消息处理。ObjectWindows Class Library 库消除了最易产生编程错误的来源，即 Windows API 的消息循环。OWL 库被设计来自动处理每一条 Windows 消息。代替使用标准 switch-case 语句，每一条 Windows 消息被直接映射到一个进行处理的成员函数。
- OWL 库允许自诊断。OWL 库实现了自诊断的能力，这就意味着可以在一种易于理解的格式下把有关多个不同对象的信息转储到文件中去，并且验证一个对象的成员变量。
- OWL 库具有一个稳定的体系结构。因为事先考虑到 ANSI C 的 throw/catch 标准，ObjectWindows Class Library 已经实现了一个扩展的异常处理体系结构。这允许一个 OWL 对象可从标准错误状态恢复过来，这些标准错误包括“内存溢出”错误、无效选项、文件或资源加载问题，在体系结构中的每个成员同 ANSI C 建议的标准能提供向上兼容。
- OWL 库提供动态对象类型。这一特别有用的性能把动态分配对象类型的确定延迟到运行时才进行。这样便可以不必担心一个对象的数据类型而进行处理。因为关于对象类型的信息在运行时被返回，程序员可以减少许多细节处理。
- OWL 库可以和谐地同以 C 语言为基础的 Windows 应用程序共存。ObjectWindows Class Library 最重要的特征是同以 C 语言为基础的使用 Windows API 的 Windows 应用程序共存的能力。在同一个程序中，程序员可以同时使用 OWL 类和 Windows API 调用。这样，就可按经验和需要在一个使用 OWL 的应用程序加进真正的C++面向对象代码。上面透明的环境成为可能是，因为在两个体系结构使用了公共的命令协议。这就意味着 OWL 头文件、类型及全局定义与相应的 Windows API 名字不冲突。透明的内存管理也是达成上面和谐关系的关键因素之一。
- OWL 库可以和 MS-DOS 应用程序一起使用。ObjectWindows Class Library 是专门用来设计 Windows 应用程序的。然而，许多类提供了使用频繁的 I/O 文件和串操作所需要的对象。因此，这些通用类既可在 Windows 应用程序中使用，也可在被 MS-DOS 应用程序中使用。

在理解每个类的成员和用法时，首先要清楚 OWL 类的体系结构，包括继承和派生关系，父类和子类之间有许多相似之处，相同的数据和函数成员；具有相同父类的子类也有许多相似之处，具有许多同名的数据和函数成员，不同的是具体的数据内容。掌握这一点，在学习和理解 OWL 时，可以减少记忆量。

Borland C++从4.0开始，是一个很好用的Windows应用程序开发环境。新的IDE(集成开发环境)可在Windows中直接调试程序，工程管理比以前更方便。

Borland C++配备了Turbo Debugger for Windows和Turbo Debugger for Win32用来调试设计16位和32位的Windows应用程序，增强了包括图标、光标、位图、对话框、菜单、热键和字体等的资源设计。特别是在Borland C++ 4.0和4.5中，资源设计已得到显著增强，用来设计Windows应用程序的界面。Borland C++在OWL(ObjectWindows Class Library)中封装了Windows API(Application Program Interface)包括函数、数据结构和消息等，可以加快读者用C++开发Windows应用程序的速度，减轻读者用C++开发Windows应用程序的难度。Borland C++能用来开发Windows、Win32s、Windows 95和Windows NT应用程序。

本书首先由浅入深地介绍了用ObjectWindows Class Library设计Windows应用程序的原理，然后详细介绍了Borland C++用于C++的ObjectWindows类库的每个函数和类。按字母顺序介绍了Borland公司C++语言产品的每个ObjectWindows类库(OWL)中的类，具体地介绍了它们的成员、其功能、用法、原型所在的头文件、所有的数据和函数成员的参考和用法。本书是使用Borland公司C++语言产品的程序员的编程参考，适合于所有用Borland C++编写应用程序的读者。

编　者

1995年1月于北京

目　　录