



C++ Builder

编程技巧、经验与实例

王小华 编著



人民邮电出版社
POSTS & TELECOM PRESS

TP312
13330

C++ Builder

编程技巧、经验与实例

王小华 编著

北方工业大学图书馆



00548716

人民邮电出版社

图书在版编目 (CIP) 数据

C++ Builder 编程技巧、经验与实例/王小华编著. —北京: 人民邮电出版社, 2004.1
ISBN 7-115-11718-7

I. C... II. 王... III. C 语言—程序设计 IV. TP312

中国版本图书馆 CIP 数据核字 (2003) 第 110763 号

内 容 提 要

本书的内容包括 C++ Builder 下的 Win32 API 应用、图像处理、DLL、组件的技巧应用、自制组件、ActiveX 组件、文件处理技巧、打印功能扩展、高级数据库技术、剪贴板应用、多媒体技术、应用软件设计技巧等。

本书从应用的角度出发, 以大量的实例, 讲述编程的方法与技巧。全书共分 16 章, 每章内归纳了若干条技巧, 每条技巧分别讲了应用的背景、目的、方法, 并给出了实例代码, 最后进行了简洁的分析和扩展。

本书适合初步掌握了 C/C++ 语言语法, 希望快速获得一些编程经验、掌握编程技巧的读者学习使用。对用 Visual C++ 编程的读者同样有参考价值。

C++ Builder 编程技巧、经验与实例

◆ 编 著 王小华
责任编辑 王文娟

◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街 14 号
邮编 100061 电子函件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
读者热线 010-67132692
北京汉魂图文设计有限公司制作
北京顺义振华印刷厂印刷
新华书店总店北京发行所经销

◆ 开本: 800×1000 1/16
印张: 24.25
字数: 506 千字
印数: 1-5 000 册

2004 年 1 月第 1 版
2004 年 1 月北京第 1 次印刷

ISBN 7-115-11718-7/TP · 3639

定价: 38.00 元 (附光盘)

本书如有印装质量问题, 请与本社联系 电话: (010) 67129223

前 言

我们在学习了一门计算机语言后，总想利用它编写出一些精彩的程序；许多年轻的读者，总希望快速地获得一些编程的宝贵经验，迅速成为“高手”。然而，精彩程序的背后是数年经验的积累。

早在 1992 年，我开始用 Borland 公司的 C++ 在 DOS 操作系统下开发应用程序，为什么要选择它呢？一个很简单的理由，喜欢它的集成开发环境，它集编程、调试、运行于一体，初学者一看就懂，一用就明白，节省了入门时间。后来，操作系统发展到了 Windows，在 DOS 环境下玩得烂熟的编程技巧失去了用武之地。这期间，也试用了许多可视化开发工具，却不太如意。后来，Borland 公司变成了 Inprise 公司，推出了 Delphi 与 C++ Builder，于是我又爱不释手地用上了这两个工具。1998 年以后，我编写了中小企业管理软件《通用财务管理系统》与《通用进销存决策管理系统》，许多程序爱好者从网上获得试用版后，纷纷发来电子邮件，询问了大量的编程方法与技巧，由于来信太多，不能一一作答。我想，等空闲下来，把这些问题集中起来，结合我十多年的编程经验，写一本关于编程技巧方面的书，来回答这些支持我的程序爱好者。虽然这一等就是好几年过去了，许多当年的初学者也已变成了“高级玩家”，同时也已有无数的新生力量补充到了编程爱好者行列，为了实现这个心愿，在牺牲了多少个休息之夜后，终于整理出了这本书。如果这本书能帮助读者多掌握一点编程方法，开阔一点编程思路，我也就心满意足了。

本书从应用的角度出发，从一个一个小问题着手，讲述 C++ Builder 各个功能的实现方法，以及实现这些功能所应注意的问题，哪些是我们应该避免和克服的，哪些是我们应该发挥和继承的。全书共分为 16 章，第 1 章讲述 C 语句的基本功，主要是帮助初学者加深对 C 语言的理解。第 2 章讲述控件的应用与改造，选取了一些我们常用的控件，重点介绍它们的用法，以及改造控件的方法。第 3 章讲述了一些关于程序与窗口方面的应用知识。第 4 章介绍了与系统有关的常用功能的实现方法。第 5 章讲述了设计安装程序与卸载程序的方法。第 6 章讲述了如何

实现剪贴板的读写功能。第 7 章讲述了文件读写、目录增减等方法。第 8 章讲述了画布、图像显示、图像变幻、艺术显示、动画实现方法、位图的 CMYK 分色技术等。第 9 章讲述了如何编写自己的打印程序，以及如何进行各种打印设置。第 10 章讲述了动态链接库 DLL 的编写与调用方法。第 11 章讲述了如何在程序中读写注册表的方法。第 12 章讲述了内存与字符串常用的功能。第 13 章介绍了 OLE 与 DDE，以及程序中如何实现与 Word、Excel 的接口。第 14 讲述了一些数据库的常用处理方法：数据源拷贝、记录的增删改、字段查询、表格控件的应用等。第 15 章讲述了与多媒体有关的知识。第 16 章介绍了网络应用方面的知识。

由于时间与篇幅所限，加上本人水平有限，不可能穷尽所有的应用方法。如果读者能从本书的一个个具体实例中，获得一些编程的灵感与启发，那将是我感到最为快慰的事情。

作者
2004.1

目 录

第 1 章 C 语言基本功	1
1.1 容易理解但不一定会灵活运用的常量、变量、表达式	1
1.2 整数、浮点、双精度，使用起来要心中有数	3
1.3 逻辑操作，很少用却很有用	4
1.4 整数分支用 switch...case	6
1.5 if{ }else if{ }else{ }——字符串判断最合适	7
1.6 for 循环，不要变成死循环	8
1.7 条件循环用 while	9
1.8 continue，让循环不做工作自己走	10
1.9 break，让循环刹车	11
1.10 指针指向何方，前途各不一样	12
1.11 Class，想说爱你不容易	13
1.12 struct，常用用它可真省事	14
1.13 数据类型之间的转换	15
第 2 章 控件的应用与改造	17
2.1 无所不在的 MessageBox 对话框	17
2.2 随手拈来 ShowMessage	19
2.3 简便易用的 InputQuery 输入框	20
2.4 Label 控件显示多行文本	20
2.5 Label 控件显示超长文本时自动换行	21

2.6	如何使 Label 控件上的文字纵向显示	21
2.7	鼠标指向 Label 控件时改变说明文字与颜色	23
2.8	Edit 作为密码输入框	25
2.9	Edit 内可不可以显示图像	26
2.10	去掉 Edit 控件回车后的提示音	26
2.11	静态与动态为 SpeedButton 按钮控件添加位图的方法	27
2.12	让 SpeedButton 按钮控件显示多行说明文本	28
2.13	SpeedButton 按钮控件中的 Flat 属性	28
2.14	StringGrid 单元格赋值	28
2.15	StringGrid 单元格字体及背景颜色设置	30
2.16	StringGrid 中文字的换行与对齐方式	31
2.17	改造 Edit 只允许输入数字	34
2.18	用 Edit 与 CSpinButton 组合制作一个数量单位输入框	36
2.19	在程序中创建 Edit	37
2.20	去掉 Edit 进入时自动选择文本功能	39
2.21	程序中用代码设置选择文本	39
2.22	在 Edit 输入回车键时跳到下一个控件	39
2.23	根据实际数量的多少动态生成 Button	41
2.24	自己创建一个带图形的 CheckBox	45
2.25	自己创建一个带图形的 RadioButton	46
2.26	获取与设定 Memo 中光标的位置	47
2.27	程序中用代码控制 Memo 中文字的滚动	48
2.28	替换 Memo 控件中默认的菜单	49
2.29	将 Memo 中的内容保存为文本文件	49
2.30	让 ListBox 显示图像列表	53
2.31	为 ListBox 添加水平滚动条	55
2.32	为 ListBox 列表框添上一幅有图案的背景	57
2.33	两个 ListBox 之间交换数据	58
2.34	改变 ListBox 列表内容的顺序	59
2.35	在 ListBox 中寻找字符串	61
2.36	让 BitBtn 按钮控件显示多行说明文本	62
2.37	动态为 BitBtn 按钮控件添加位图	63
2.38	获取 BitBtn 按钮控件中位图的颜色值	65

2.39	为 GroupBox 控件加上图形背景	67
2.40	为 RadioGroup 控件加上图形背景	69
2.41	Combox 中增加自动搜索功能	70
2.42	利用 ProgressBar 显示程序运行中的进度	71
2.43	利用 CGauge 显示程序运行中的进度	74
2.44	利用 ScrollBox 实现应用程序在屏幕上的滚动	75
2.45	动态设定 ScrollBox 的水平与垂直滚动条	76
2.46	为 ScrollBox 画上图案背景	77
2.47	TreeView 中添加项目	78
2.48	TreeView 中添加图形	81
2.49	获取 TreeView 中的选择项目值	83
2.50	对 TreeView 中的项目进行排序	84
第 3 章	应用程序与窗口	85
3.1	自定义应用程序的图标	85
3.2	改变应用程序的光标	86
3.3	让应用程序脱离支持文件而独立运行	87
3.4	窗口的初始化	89
3.5	闪烁窗体的标题栏	89
3.6	让窗体的标题栏文字一个接一个显示	90
3.7	窗体定时关闭的小方法	91
3.8	关闭多余窗口以减少内存开销	94
3.9	建立 MDI 子窗口	96
3.10	关闭 MDI 子窗口	97
3.11	获取 MDI 所有子窗口	97
3.12	打开一个新 MDI 子窗口时关闭其他 MDI 子窗口	97
3.13	为应用程序制作一个活动的图标	98
3.14	隐藏桌面的图标	100
3.15	隐藏 Windows 开始菜单的方法	101
3.16	夺取程序的控件权	101
3.17	用按钮关闭模式窗口的方法	102
3.18	利用程序代码关闭模式窗口的方法	104
3.19	让窗口永远显示在最前面	104

3.20	解决浮点运算中的四舍五入问题	105
3.21	避免自己的应用程序二次运行	106
3.22	怎样关闭别的应用程序	107
3.23	如何模拟键盘输入	109
3.24	截获窗体的关闭信息以阻止窗体关闭	110
3.25	按下 Esc 键退出程序	111
3.26	如何制作透明的窗口	113
3.27	程序执行后自己删除自己	113
3.28	枚举 Windows 系统中的字体	114
3.29	巧用控件中的 Tag 属性	116
3.30	巧妙利用文本文件保存应用程序的菜单内容	118
第 4 章	系统	125
4.1	如何处理 Windows 应用程序的命令行参数	125
4.2	文件的 DOS 日期转换为 TDate	126
4.3	格式化软盘	127
4.4	利用软盘序列号的加密方法	128
4.5	不让应用程序显示在任务栏上	131
4.6	在程序中利用 WM_SYSCOMMAND 消息启动屏幕保护程序	133
4.7	获取 Windows 的系统相关目录	135
4.8	如何制作托盘图标	136
4.9	执行控制面板上的程序	139
4.10	修改系统日期	142
4.11	获取 Windows 系统信息	143
4.12	调整显示分辨率	144
4.13	检测磁盘的容量	146
4.14	检测磁盘的剩余空间	147
4.15	在程序中用代码关闭 Windows 系统	148
4.16	删除文件至回收站	149
4.17	程序中如何更改 Windows 桌面	149
第 5 章	应用程序的安装与卸载	153
5.1	如何设计自己的安装程序	153

5.2	软件序列号设置的方法	154
5.3	将应用程序放在桌面上	155
5.4	卸载程序如何实现	159
5.5	OCX、DLL 文件的注册方法	162
5.6	文件的压缩与解压	164
5.7	如何将应用程序加入启动	168
5.8	如何将文件分割	169
5.9	如何组合分割后的文件	173
第 6 章	剪贴板	177
6.1	监视剪贴板的内容	177
6.2	文本的复制与粘贴	178
6.3	图像的复制与粘贴	180
6.4	流与剪贴板	181
第 7 章	文件与目录	183
7.1	文件指针的活用	183
7.2	如何在目录中搜索文件	185
7.3	自制文件列表并按类型显示位图	188
7.4	INI 文件的读写	192
7.5	获取驱动器类型	194
7.6	删除隐含文件的方法	196
7.7	删除只读文件的方法	198
7.8	文件拷贝	198
7.9	文件删除	200
7.10	创立文件夹	200
7.11	一次性创建多层目录	201
7.12	删除文件夹	202
7.13	获取文件的日期	203
7.14	修改文件的日期	205
7.15	长文件名转换为短文件名	207
7.16	只更改文件的扩展名	208
7.17	将目录删除至回收站	209

7.18 复制目录树	209
第 8 章 画布与图形图像	211
8.1 RGB 与 TColor 的转换	211
8.2 TCanvas 与 Font	212
8.3 移动动画的实现	213
8.4 利用 ScrollBox 滚动显示大位图	219
8.5 实现屏幕拷贝至剪贴板	220
8.6 如何改变画布的文字显示分辨率	221
8.7 如何改变画布的分辨率	223
8.8 不同分辨率的画布画同样比例的图形	225
8.9 位图的 CMYK 分色技术	227
8.10 图像的放大与缩小	230
8.11 艺术显示图片	231
8.12 BMP 与 JPG 格式转换	236
8.13 图像翻转	237
8.14 获取图像中的颜色值	239
8.15 防止图像显示的闪烁现象	240
8.16 图像的分层处理方法	242
8.17 将位图的二进制数转化为文本	244
8.18 将彩色位图变为灰度位图	248
8.19 在桌面上画图	249
8.20 如何实现文本自适应显示区域	250
第 9 章 打印	253
9.1 认识 TPrinter	253
9.2 打印画布	256
9.3 简单文本打印	256
9.4 位图打印	259
9.5 打印纸张设置	260
9.6 打印纸张横向与纵向调整	265
9.7 打印纸张的定位方法	266
9.8 打印比例设置	267

9.9 设置打印颜色	269
9.10 设置打印质量	269
第 10 章 动态链接库 DLL	271
10.1 创建 DLL	271
10.2 静态调用 DLL 的步骤	273
10.3 动态调用 DLL 的步骤	274
10.4 DLL 中的 Form	276
10.5 DLL 入口与出口—— DllEntryPoint	279
第 11 章 注册表	282
11.1 设置注册表	282
11.2 通过注册表获取 Windows 的信息	284
11.3 通过注册表设置文件的关联执行程序	286
11.4 通过注册表查找扩展文件名的关联执行程序	287
11.5 巧用注册表保存应用程序的运行状态	288
11.6 通过注册表查找打印机安装信息	289
11.7 通过注册表查找显示器信息	290
11.8 巧妙利用注册表保存自己的密码	291
第 12 章 内存与字符串	293
12.1 去掉字符串的空格	293
12.2 字符串大写与小写转换	294
12.3 字符串比较	295
12.4 找出字符串中不含另一字符串字符的字符	297
12.5 字符串的位加密方法	298
12.6 查找与替换字符串中的内容	300
12.7 字符串截取	302
12.8 判断汉字的内码	302
12.9 动态内存分配方法	303
12.10 利用内存流读位图文件	307
12.11 利用内存流将小文件合并成大文件	308

第 13 章 OLE 与 DDE	310
13.1 OLE 容器控件	310
13.2 编辑 OLE 控件	311
13.3 存取 OLE 对象	312
13.4 与 Word 的接口	313
13.5 与 Excel 的接口	314
13.6 Word 文档用数据库管理	316
13.7 DDE 客户端程序设计	320
13.8 DDE 服务器端程序设计	322
第 14 章 数据库	324
14.1 自己在程序中配置 ODBC 数据源	324
14.2 创建数据表	326
14.3 修改 Query 的查询结果集	328
14.4 利用 Query 实现 Pack 功能	328
14.5 利用 BatchMove 实现 Pack 功能	331
14.6 查找的 Locate 方法	333
14.7 查找的 Query 方法	334
14.8 Query 实现模糊查询	335
14.9 Table 添加记录的方法	336
14.10 Query 添加记录的方法	337
14.11 Table 修改记录的方法	338
14.12 Query 修改记录的方法	339
14.13 Table 删除记录的方法	340
14.14 Query 删除记录的方法	340
14.15 记录批量增加	341
14.16 记录批量修改	341
14.17 记录批量删除	342
14.18 如何获取数据库中的表名	343
14.19 如何获取表中的字段名	343
14.20 求记录中字段的最大值	343
14.21 记录中的字段求和	344
14.22 记录中的字段求平均	345

14.23	(TDate)0 是什么东西	346
14.24	甩掉报表控件, 自己制作的报表更好看	347
第 15 章	多媒体	356
15.1	利用 TAnimate 制作动画	356
15.2	利用 TTimer 制作动画	358
15.3	检测声卡是否存在	360
15.4	检测光驱中是否有 CD	360
15.5	弹开与关闭光驱	361
15.6	禁止与启用光驱的自动播放功能	362
15.7	电影全屏播放	363
15.8	伴音的实现	365
第 16 章	网络	367
16.1	获取电脑所处的工作组名	367
16.2	获取并修改计算机名	368
16.3	如何设置网络驱动器映射方式	369
16.4	如何在程序中链接自己的网页和邮件	370
16.5	测试是否联网	371
16.6	在程序中启动拨号	372

第 1 章 C 语言基本功

1.1 容易理解但不一定会灵活运用常量、变量、表达式

所谓常量，即在程序运行过程中，以一个固定的值表现的量，在 C 语言中，用关键字 `const` 声明。对于常量，在使用时，注意以下两点。

1. 常量在声明时就要赋值。

如 `const int abc=1234;`

2. 常量的属性是只读的，在使用过程中，不能再被赋值。

如 `abc` 在定义时已被赋值为 1234，如果在程序中再用这样的语句“`abc=22;`”系统在编译时将不被通过。

常量在什么时候能用到呢？举一个简单的例子：在数学的几何运算中，常常遇到圆周率，它的值是 3.1415926，在运算时，有时需要多次调用这个值，而且这个值的小数位特别长，稍不留意就会写错。为了不用每次都写这个值，常量的作用就发挥出来了。我们用一个常量定义“`const double PI=3.1415926;`”以后，程序中凡有圆周率的地方，就都可以用 `PI` 代替了。

变量则与常量相反，它是一个变化的量，在程序设计中，可以根据实际情况随时赋予它不同的值。变量由数据类型声明。

定义一个整数变量，声明格式为：

```
int aa;
```

定义一个双精度型的变量，声明格式为：

```
double aa;
```

定义一个标准字符串型的变量，声明格式为：

```
AnsiString aa;
```

变量可以在声明的同时赋值，如：

```
int aa=123;
```

在值不确定的情况下，可以不赋值。

变量根据声明的位置不同，可以分为全局变量与局部变量。

所谓全局变量，有一个简单的区分方法：即声明于函数之外的为全局变量，该变量可应用于每一个函数。声明于函数内部的为局部变量，该变量仅限于本函数使用。

最常犯的错误是将全局变量与局部变量取同一个名称，这样，在程序编译时，不会发生任何错误，但是，在程序调试时，却会出现莫名其妙的错误，如果没有意识到同名的问题，无论怎么寻找，就是找不出程序的错误。

谁都不能保证自己曾经声明过的变量会永远记得，可以通过一些手段来弥补记忆的不足。比如，凡是全局变量，在变量名前加上标识符，常用的标识符为“G_”，如声明某些变量如下：

```
int G_aa1,G_aa2,G_Width;
```

对于局部变量的声明，可以在函数的任何地方，如：

```
void Function1(void)
{
    int aa;
    aa=123;
    int bb;
    bb=234;
    int cc;
    cc=aa*bb;
}
```

这是一个简单的函数，用到了3个变量，每一个变量都是在快要用到时才声明，这种写法在早期的C语言中是不允许的，但随着C语言的升级，现在允许了。但这是不是最好的方式？从经验角度来看，并不合适：

1. 程序不直观，不能一下子看出用了多少个变量，在检查程序时，容易产生很多误解。
2. 容易造成变量堆积，不能有效和充分地使用变量，不利于程序优化。

对于上面的简单函数，可以写成下面的形式：

```
void Function1(void)
{
    int aa,bb,cc;
    aa=123;
    bb=234;
```

```
cc=aa*bb;
}
```

别以为这是个习惯问题，也许就是这个习惯，导致在查找程序错误时头疼不已。

很多程序员往往不太喜欢使用变量，尤其不喜欢将变量写得太长，嫌麻烦。比如有些人总是喜欢用 a、b、c、i、j、aa、bb、cc、ii、jj 等简单的字母表示。当然，在编写一些小程序时，一共也用不了几个变量，很容易将变量的意思分清，用简单字母表示也未尝不可。但要是大程序呢？上千个变量甚至上万个变量呢？还能保证永远记住 i 与 j 这些变量是代表什么吗？

不要贪图简单而给自己程序检查时设置障碍，要知道，很少有一次编写成功的程序，一个程序往往要经过无数次的修改、补充才能达到要求。如果设置的变量不能清楚地表达意思，每一次检查与修改时，都要重新搞清楚各变量的意思，将会非常费时。

在设置变量名时，特别是全局变量名时，要尽可能地将变量的意思体现出来，不要怕名字太长。例如定义表的宽度变量，如果习惯英语，就用英文 Table_Width 表达，如果习惯汉语拼音，就用 Biao_Kuan 表达，这样会一目了然。如果用 a 表达，三天以后谁能保证还记得它表达的意思。

如果变量所要表达的意思很多，尽可多用“_”符号，将不同的意思连结起来，便于自己理解，也便于别人能看懂。

表达式是常量、变量、运算符的组合，至于如何写表达式，这里介绍几点应该注意的方面。

1. 表达式不宜写得太长，对于运算关系复杂、运算符多、括号多的表达式，应该分开来写，不要图省事，尽可将一行拆分为多行来写。如某表达式：

```
aa=bb*(1+b)*(2+c+3*(3+g));
```

可以写为：

```
aa=bb*(1+b)
```

```
aa=aa*(2+c+3*(3+g));
```

2. 对于有浮点运算的表达式，要注意采用双精度型，即用 double 声明的类型。类型不同的变量进行运算，要注意类型转换。

1.2 整数、浮点、双精度，使用起来要心中有数

在数学运算中，整数、浮点数、双精度数是常用到的，这里不讲概念，谈几点经验。

1. 要尽可能多运用整数进行运算，因为它运行的速度快。现在的计算机发展了，也许感觉不出整数与浮点数之间运行速度的差别，但在处理大数据量的运算或大型数据库时，区别就很明显了。有时为了提高运算速度，有经验的程序员甚至将浮点数转化为整数运算。如某浮点数为 12.34，我们将它乘以 100，变为整数 1234，然后再参与运算。