C O M P U T E R

- 为什么要使用程序来维护注册表
- 编程维护注册表的基本方法和实现方法
- 程序读取注册表应用实例
- 维护注册表的专用C语言类



ISBN 7-900335-37-4

光盘含：源程序代码

©装帧设计\何东琳设计工作室
ISBN 7-900335-37-4
定价：29.00元

需要全本请访问 www.ertongbook.com

中文 Windows 2000 注册表编程最佳指南（光盘配书）

浦东电子出版社 Pudong Electronic Press

C O M P U T E R

Windows2000系列丛书

中文 Windows2000 注册表编程最佳指南

ZHONGWEN Windows2000 ZHUCEBIAO
BIANCHENG ZUIJIA ZHINAN

◎ 张伟锋 龙法宁 编著



浦东电子出版社 Pudong ePress

中文Windows2000
注册表编程最佳指南



张伟锋 龙法宁 编著

浦东电子出版社
pep pudong epress

内容简介

本书介绍了利用 Visual C++ 6.0 编写有关 Windows 注册表操作的应用程序的技术，内容包括注册表编程的基本操作和一些高级的应用技巧，以及如何利用注册表编程辅助普通应用程序的编程，如何利用注册表提高应用程序的性能。此外，本书还介绍了大量与编程有关的注册表知识。

本书内容实用，技术先进，范例典型，语言通俗易懂。

本书既可作为 Visual C++ 6.0 的深入学习的教材,也可以作为开发 Windows 注册表应用程序的经典书籍。适用于 Windows 应用程序开发人员和计算机爱好者。

光盘中包括本书重要实例的完整代码和可执行文件，配有与注册表相关的重要附录。

书　　名：中文 Windows 2000 注册表编程最佳指南
文 本 著 作 者：张伟锋 龙法宁
C D 制 作 者：本社多媒体研究制作中心
责 任 编 辑：廖果 黄益斌
出版、发行者：浦东电子出版社
地　　址：上海浦东郭守敬路 498 号上海浦东软件园内　201203
电话：021-38954510，38953321，38953323（发行部）
经　　销：各地新华书店、软件连锁店
排　　版：四川中外科技文化交流中心排版制作中心
C D 生 产 者：湖南省远景光电实业有限公司
文 本 印 刷 者：郫县犀浦印刷厂
开 本 / 规 格：787×1092 毫米　16 开本　14.875 印张　240 千字
版 次 / 印 次：2001 年 1 月第一版　2001 年 1 月第一次印刷
印　　数：0001——8000 册
本 版 号：ISBN7—
定　　价：28.00 元（1CD，含配套书）

说明：凡我社光盘配套图书有缺页、倒页、脱页、自然破损，本社发行部负责调换。

目　录

第一章　注册表与编程 …… 1
1.1 注册表、Windows 操作统、应用程序三者之间的关系 …… 1
1.1.1 注册表与Windows 的关系 …… 1
1.1.2 Windows 操作系统与应用程序的关系 …… 1
1.1.3 注册表与应用程序的关系 …… 1
1.2 应用程序与Windows 整合与注册表项目的添加 …… 2
1.3 注册表的修改方法 …… 2
1.3.1 手动方式 …… 2
1.3.2 安装制作软件方式 …… 3
1.3.3 编程方式 …… 4
1.3.4 编程实现的意义 …… 5
1.4 本章小结 …… 5
第二章　注册表结构概述 …… 6
2.1 注册表的由来 …… 6
2.2 注册表的作用 …… 6
2.3 注册表操作的注意事项 …… 6
2.4 注册表的文件组成 …… 7
2.5 注册表的组织形式 …… 8
2.6 数据类型 …… 9
2.6.1 各根键之间的相互关系 …… 9
2.6.2 注册表键值项数据类型 …… 10
2.6.3 注册表数据类型 …… 10
2.6.4 注册表与INI 文件的关系 …… 11
2.7 注册表的修复 …… 13
2.8 本章小结 …… 13
第三章　注册表编程常用技巧举例 …… 14
3.1 程序的自启动 …… 14
3.2 将网页作为默认主页 …… 15
3.3 在浏览器的标准按钮中添加按钮 …… 16
3.4 本章小结 …… 19
第四章　MFC 注册表函数操作实例 …… 20
4.1 MFC 的自带注册表函数介绍 …… 20
4.2 MFC 注册表函数操作实例 …… 21
4.2.1 建立项目 …… 21
4.2.2 修改代码 …… 21
4.2.3 处理WM—CLOSE 消息 …… 22
4.2.4 处理PreCreateWindow() 函数 …… 24
4.2.5 代码分析 …… 25
4.3 本章小结 …… 26
第五章　注册表的API 操作函数 …… 27
5.1 内容简介 …… 27
5.1.1 打开,创建和关闭一个子键 … 28
5.1.2 写入和删除注册表数据 …… 28
5.1.3 注册表数据的恢复 …… 28
5.1.4 注册表文件 …… 28
5.2 注册表修改实例 …… 28
5.2.1 更改IE 背景并使程序自启动 …… 28
5.2.2 注册表的查询与修改 …… 36
5.3 本章小结 …… 42
第六章　连续字符串的注册操作 …… 43
6.1 内容提要 …… 43
6.2 方法介绍 …… 43
6.2.1 对连续的字符串写入的操作函数 …… 44
6.2.2 对连续的字符串读出的操作函数 …… 45
6.2.3 对特定的已打开主键下的所有子键读出的操作函数 …… 47
6.3 运行结果 …… 49
6.4 本章小结 …… 50
第七章　注册应用程序的数据文件 …… 51
7.1 注册文件类型 …… 51
7.2 程序实现的具体思路 …… 52
7.3 编程步骤 …… 52
7.4 本章小结 …… 55
第八章　注册表封装类CRegistry …… 56
8.1 内容简介 …… 56
8.2 使用方法 …… 56
8.3 封装类源代码 …… 56
8.4 具体使用方法 …… 83
8.4.1 创建子键代码 …… 83
8.4.2 删除子键的操作代码 …… 83
8.4.3 注册表读操作 …… 84

8.4.4 注册表写操作 …… 85
8.5 CRegistry 公有变量说明 …… 90
8.6 本章小结 …… 92
第九章 注册表实现IE 的功能 …… 93
9.1 收藏夹功能的实现 …… 93
9.1.1 框架处理的代码添加及代码分析 …… 93
9.1.2 BuildFavoritesMenu 函数的代码添加及代码分析 …… 96
9.2 IE 地址栏组合框的实现 …… 99
9.2.1 工具条初始化代码添加及分析 …… 99
9.2.2 成员AddressInit 函数代码的添加与分析 …… 103
9.3 本章小结 …… 109
第十章 注册表修改实例 …… 110
10.1 配置启动和登录选项 …… 110
10.1.1 Windows 2000 启动过程 … 110
10.1.2 在启动时显示提示信息 …… 111
10.1.3 禁止启动时的警告信息 …… 112
10.1.4 管理启动时的驱动器共享 …… 113
10.1.5 自动重新启动外壳程序 …… 114
10.1.6 启动时使用或禁用Autoexec.bat …… 115
10.1.7 更改登录时的背景图案 …… 116
10.1.8 设置有效密码 …… 117
10.1.9 自动登录 …… 118
10.2 配置关闭选项 …… 120
10.2.1 设置自动关机 …… 120
10.2.2 设置关闭系统之前自动结束任务 …… 121
10.3 配置开始菜单 …… 122
10.3.1 添加“收藏夹”菜单项 …… 122
10.3.2 添加“注销”菜单项 …… 123
10.3.3 添加“管理工具”菜单项 …… 124
10.3.4 删除“Windows Update”菜单项 …… 125
10.3.5 删除“文档”菜单 …… 127
10.3.6 清空“文档”菜单 …… 128
10.3.7 删除“收藏夹”菜单项 …… 130
10.3.8 删除“搜索”菜单项 …… 131
10.3.9 删除“运行”菜单项 …… 133
10.3.10 删除“网络和拨号连接”菜单项 …… 134
10.3.11 级联特殊的文件夹 …… 135
10.3.12 禁止修改“开始”菜单 …… 137
10.3.13 禁用任务栏的快捷菜单 … 139
10.3.14 禁止显示提示信息 …… 140
10.3.15 禁用智能菜单 …… 141
10.3.16 在“运行”对话框中添加选项以便安全运行16 位应用程序 … …… 142
10.4 配置桌面 …… 144
10.4.1 删除“Internet Explorer”图标 …… 144
10.4.2 设置桌面墙纸 …… 145
10.4.3 设置墙纸的排列模式 …… 146
10.4.4 设置墙纸的风格 …… 147
10.4.5 激活屏幕保护程序 …… 148
10.4.6 设置屏幕保护程序的启动时间 …… 149
10.4.7 设置屏幕保护程序的运行程序 …… 149
10.4.8 设置屏幕保护程序的密码 …… 151
10.4.9 设置任务切换的行数 …… 151
10.4.10 设置任务切换的列数 …… 152
10.4.11 设置窗口拖动方式 …… 154
10.4.12 设置拖动检测宽度 …… 154
10.4.13 设置拖动检测高度 …… 156
10.4.14 设置菜单延时 …… 157
10.4.15 设置平滑字体边缘 …… 158
10.4.16 设置鼠标方向轮的滚动行数 …… 159
10.5 配置控制面板的显示 …… 160
10.5.1 禁用控制面板 …… 160
10.5.2 删除控制面板中指定的功能图标 …… 161
10.5.3 显示控制面板中指定的功能图标 …… 165
10.6 配置“显示”图标 …… 168
10.6.1 禁用“显示”功能图标 …… 168
10.6.2 删除“背景”标签 …… 170
10.6.3 删除“屏幕保护程序”标签 …… 171

需要更多请访问 www.ertongbook.com

10.6.4 删除“外观”标签 …… 173
10.6.5 删除“设置”标签 …… 174
10.6.6 禁用“背景”标签 …… 175
10.7 配置键盘选项 …… 177
10.7.1 设置键盘延时 …… 177
10.7.2 设置键盘重复率 …… 178
10.7.3 设置光标闪烁频率 …… 179
10.8 配置文件夹选项 …… 180
10.8.1 配置“文件夹选项”对话框的界面 …… 180
10.8.2 设置记住每个文件夹的视图位置 …… 183
10.8.3 设置使用交替的颜色显示压缩的文件和文件夹 …… 184
10.8.4 设置隐藏受保护的操作系统文件 …… 185
10.8.5 设置显示所有文件和文件夹 …… 186
10.8.6 设置隐藏已知文件类型的扩展名 …… 186
10.8.7 设置在单独的进程中打开文件夹窗口 …… 187
10.8.8 设置允许在文件夹中使用Web 内容 …… 188
10.8.9 设置在标题栏中显示全路径 …… 190
10.8.10 设置在地址栏中显示全路径 …… 191
10.9 配置资源管理器 …… 192
10.9.1 删除“文件”菜单 …… 192
10.9.2 显示“映射网络驱动器”按钮 …… 193
10.10 配置命令提示符窗口 …… 195
10.10.1 设置光标大小 …… 196
10.10.2 设置显示选项 …… 196
10.10.3 设置命令记录缓冲区大小 …… 197
10.10.4 设置命令记录缓冲区数目 …… 198
10.10.5 设置编辑选项 …… 199
10.11 配置文件打开与关联方式 …… 200
10.11.1 在文件图标的快捷菜单中添加菜单项 …… 200
10.11.2 在特定类型的文件图标的快捷菜单中添加菜单项 … 203
10.12 提高系统性能 …… 206
10.12.1 关闭安装光盘的自动播放功能 …… 206
10.12.2 自动刷新窗口 …… 206
10.12.3 自动重新启动 …… 209
10.13 本章小结 …… 210
第十一章 Office 2000 配置信息 …… 213
11.1 Office 2000 的公用注册表信息 …… 213
11.1.1 Office 2000 的版本号 …… 213
11.1.2 查看Office 2000 的安装位置 …… 214
11.1.3 查看Office 2000 的相关信息 …… 214
11.1.4 查看助手的信息 …… 215
11.2 Word 的注册表信息 …… 216
11.2.1 查看路径变量 …… 216
11.2.2 查看转换选项 …… 217
11.3 本章小结 …… 217
第十二章 注册表中的网络设置 …… 218
12.1 配置远程访问 …… 218
12.1.1 设置连接之前电话铃响的次数 …… 218
12.1.2 设置身份验证时允许重试的次数 …… 218
12.1.3 设置身份验证时的时间限制 …… 218
12.1.4 设置回拨时间 …… 219
12.1.5 设置自动断开的时间限制 …… 219
12.2 配置TCP/IP 协议 …… 219
12.2.1 查看DHCP 服务器名 …… 219
12.2.2 查看域名 …… 219
12.2.3 查看主机名 …… 219
12.2.4 查看DNS 服务器的IP 地址 …… 220
12.2.5 设置启用备份网关 …… 220
12.2.6 禁止访问TCP/IP 协议的高级选项 …… 220
12.2.7 设置超时错误的等待时间 …… 220
12.2.8 设置检测死锁网关 …… 220

12.3 配置网络打印机 …… 221
12.3.1 禁止浏览网络打印机…… 221
12.3.2 禁止删除打印机…… 221
12.3.3 禁止添加打印机…… 221
12.4 本章小结…… 221
第十三章 查看注册表中的重要信息…… 222
13.1 查看配置单元及其对应的文件…… 222
13.2 查看环境变量…… 222
13.3 查看显卡的信息…… 223
13.4 查看网卡的信息…… 225
13.5 查看鼠标的信息…… 227
13.6 本章小结…… 229



第一章 注册表与编程

1.1 注册表、Windows 操作系统、应用程序三者之间的关系

1.1.1 注册表与 Windows 的关系



注册表包含了 Windows 系统和用户程序的各种所需数据及项目配置、硬件信息等许多信息，是 Windows 的核心数据库。如果计算机以 Windows 作为操作系统，那么可以说系统的一切运行都离不开注册表。注册表控制着 Windows 操作系统的运行情况，修改注册表的某些项目可以达到优化系统、加速程序运行等目的。

1.1.2 Windows 操作系统与应用程序的关系

一台计算机能够被人操作，只有裸机是不行的，它还需要不同的软件来实现某些功能。根据任务不同，将软件环境分成几个不同的层次。操作系统属于第一层用户与机器之间的交流中间层，用户通过操作系统向计算机发出控制命令、提供数据；同样计算机通过操作系统接收用户的控制命令及数据。操作系统仅仅完成如内存分配、I/O 管理等与硬件运行密切相关的任务。在操作系统的基础上还有应用程序，如各种文字处理、图像处理等软件，这些软件就构成了应用软件层，它们往往适用于不同的领域，但功能却非常强大。

操作系统属于软件系统的最底层，直接与硬件相联系，它除了充当用户与硬件的交互作用之外，还管理应用程序的运行。尽管如今的 Windows 操作系统集成了部分应用程序，如记事本、计算器等，但这些软件从功能上只是满足一般的需求，远远不能适用于特殊的环境和满足特殊的要求。而且各个领域对于应用程序达到的目的和所要求的功能都不是完全一致，都有自己不同的需求，因此要将所有的应用程序集成到操作系统中是不可能的。

1.1.3 注册表与应用程序的关系

普通的 DOS 程序或者 Win 3.x 程序尽管仍然能够在 Windows 下运行，然而它们的速度却比 Windows 自带的一些应用程序（如记事本、计算器等）运行的速度慢了不少。同样也比专门为 Windows 设计的程序慢很多，是什么原因导致的呢？实际上，这跟注册表有着密切的关系，事实上，Windows 就是通过注册表将应用程序与自己整合在一起。整合后，操作系统与应用程序就如一个整体，操作系统将可以根据注册表中的信息为应用程序设置适当的环境变量，以使它们更好地运行。

因此，当今的流行软件，尤其是商业软件在程序安装的过程中就将自身的一些信息添加到 Windows 的注册表中，以达到整合的目的。可以说注册表是应用程序与 Windows 操作系统形成一个整体的桥梁，因此随着系统中安装的应用程序越来越多，Windows 的注册表会变得越来越大。然而，当系统卸载应用软件时，为了便于管理，Windows 操作系统只在注册表中进行删除标记，而不会真正从注册表中将与应用程序相关的项目进行清除。这样随着时间的推移，Windows 的注册表中会保留很多无用的项目，这会影响系统的运行速度。

1.2 应用程序与 Windows 整合与注册表项目的添加

一般的软件要与 Windows 整合，要做的工作有很多，典型操作有：

1.文件关联

将应用程序与要使用的文件联系在一起，这样，当完成文件关联后，用户只要点击文件，Windows 就会以相关的应用程序将此文件打开。

2.快捷方式的添加

所谓快捷方式是快速启动程序或打开文件或文件夹的方法，而不必转入“Windows 资源管理器”中它们的永久位置。快捷方式对经常使用的程序、文件和文件夹非常有用。应用程序的快捷方式至少要包含在“开始菜单”中，还可以添加在“桌面”或者“快速启动”栏中。



3.启动项目的添加

可以使程序在 Windows 启动时自动运行应用程序，这并非每个程序均要求。

以上仅为例子，根据应用程序的功能要求等不同，添加的项目也有所不同，不能一概而论。然而，所有相关的项目均与注册表的修改与添加是分不开的。

上面的例子可以通过视觉体会到，并且在注册表中都会有相应的变化，但并不是所有的注册表变化都可以通过视觉感受到。有一些仅仅保存了应用软件的信息、运行时的环境变量等，如应用程序的作者信息、版本信息、用户信息等，在程序安装时同样会装入注册表中，尤其是商用软件更是如此。有的软件甚至选择注册表来保存应用软件的序列号等信息。

1.3 注册表的修改方法

这里所要介绍的注册表修改方法主要针对新软件在注册表中的注册。因此注册表的修改方法主要有三种：一种是手动，通过 Regedit 等注册表编辑器来修改；一种是安装制作软件；最后一种是通过编程来实现。

1.3.1 手动方式

如今注册表的修改工具种类繁多，功能也相当强。最常用的还是 Windows 自带的 Regedit，采用这种方式，必须把所要添加的项目的位置找到，然后进行修改。因此比较繁杂，而且由于位置比较多，容易产生错误，发现错误也比较困难。从另一个角度来看，一个程序员编制软件后肯定会考虑让用户方便地使用它，而在安装时就有繁杂的工作要做，恐怕不是程序员的初衷，用户也不容易接受。

幸好 Regedit 还提供了一个功能，那就是注册表的导入，即将一个注册表项目文件（.reg）自动送入注册表中。.reg 文件是可以用纯文本编辑器编辑的。将下面的几行语句写入记事本并将此文件保存为.reg 文件，此时如果用鼠标点击此文件会出现提示框，询问用户是否将文件中的内容添加到注册表中，如图 1-1 所示。

```
REGEDIT4

[HKEY_LOCAL_MACHINE\Network]
```



```
[HKEY_LOCAL_MACHINE\Network\Logon]
"username"="weivan"
"PrimaryProvider"="Microsoft 友好登录"
"LMLogon"=hex:00,00,00,00
```



图 1-1

导入后的注册表会在相应的项目中添加内容，如图 1-2 所示。

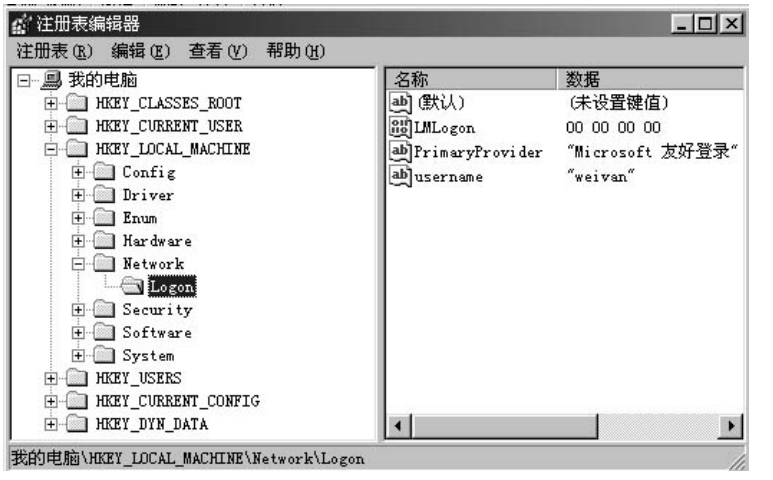


图 1-2

根据文本中的内容，其中 Network 中的内容为空，Logon 中的内容在窗口的右边显示出来。

这种方式将所有的相应内容在一个文档中列出，容易检查，在导入之前可以经过多次检查以保证注册表添加修改的项目的正确性。并且在注册表导入后，如果发现错误，还可以根据文本内容在注册表中找到相应位置进行适当的修改。这种方法似乎比原来的方法有所改进，事实上这两种方式是一致的，而且经常采用。

1.3.2 安装制作软件方式

当程序编制完成后，许多程序员比较喜欢采用安装程序将软件制作成一个安装软件包的方式发布。这种方式是目前 Windows 应用程序普遍采用的方式，也是一种比较方便的方式。尤其是对于用户来讲，早就熟悉了这种方式，用户一旦拿到一个软件，首先找的就是它的安装文件（setup.exe）。目前的安装打包程序比较多，如 Visual Studio 中就可以选择安装一个

InstallSheld 的组件，程序员可以用这个组件将应用程序打包成安装软件包发布。在制作安装软件包时，可以将要修改的注册表项目添加入注册表中。

Visual Studio 的 InstallSheld 安装文件制作组件功能非常强大，然而它的使用也相当复杂，其中添加注册表等采用编程实现，所使用的语言与 Visual C++语言有点差异，这使得制作安装文件变得非常困难。采用编程方式，它的灵活性比较高，可以完全按照程序员的要求定制安装程序。

目前，安装制作软件的品种相当丰富，功能也不是很弱，如 Gksetup、Wise setup 等软件。这些安装制作软件的优点就是界面全可视化，无需采用编程方式。它的缺点就是灵活性不够高，程序员并不能完全根据自己的要求定制安装程序，必须跟着软件一步步完成。因此这种软件功能不是非常强大，但对于小型软件已经足够了。此类软件中添加注册表项目同样采用可视化界面，图 1-3 就是在 Gksetup 中进行注册表项目添加时的窗口，这使得添加与进行注册表修改变得异常容易。



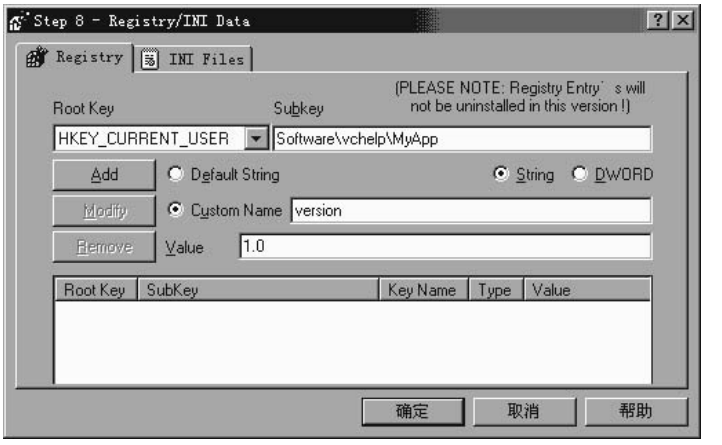


图 1-3

然而这种安装制作软件的注册表添加方式从本质上来讲与方式一是一样的，当然它采用了图形界面，与用户的交互比较友好。用户均希望在软件卸载的时候，卸载程序能够将注册表中与此软件相关的内容从注册表中删除，这些方便的小程序不能达到这方面的要求。

1.3.3 编程方式

编程方式是指在进行应用程序设计、编写的时候就已经将注册表的修改项目考虑了，并且以编写代码的形式对注册表进行操作。因此注册表的修改与应用程序两者是完全紧密地联系在一起的。采用编程方式，主动权完全掌握在编写应用程序的程序员手中，因此相当灵活、并且所能实现的功能相当多。当然它和前面两种方式所面临的问题一样，程序员必须对注册表有充分的了解。如果要达到更多的效果就必须对注册表深入研究，不能只停留在了解的层次，至少应达到理解的层次。

当然，编程方式并不要求从最基本的代码开始编写。在 Visual C++中，基本上与注册表相关的操作均有对应的现成函数可以调用。

需要全本请访问 www.ertongbook.com

1.3.4 编程实现的意义

编程方式还有其他方式所不能实现的功能。由于其他方式均为一次性写入型，不能根据程序运行环境、要求等的改变进行调整。而在编写应用程序的过程中，尤其是一些比较大的软件，总会带有一个“系统设置”，作为应用程序参数改变的依据。很多这方面的参数均会与注册表有联系，那么程序员是如何实现的呢？用上面提到的两种方法显然不行，因此只有靠程序员的编程来实现。

例如有的程序可以根据用户的选择来决定在 Windows 启动时是否自启动程序。通过程序的运行就可以达到这一目的。事实上要实现程序的自启动相当容易，只要在注册表中相应位置添加相应的信息即可，如图 1-4 所示，只要将应用程序的位置信息添加到

HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\Current\Run

栏中就可以实现，在图 1-4 中的右窗口中就展示了当前系统在 Windows 启动时自动加载的项目。这样的功能实现，程序通常采用的方式即为编程方式，而用户只要在应用程序的“系统设置”中将相应的复选框选中即可。试想一下用前两种方式来实现这一功能有多麻烦，而且用户也不容易接受这样的软件，每次改变设置还须用户自己进行注册表修改。

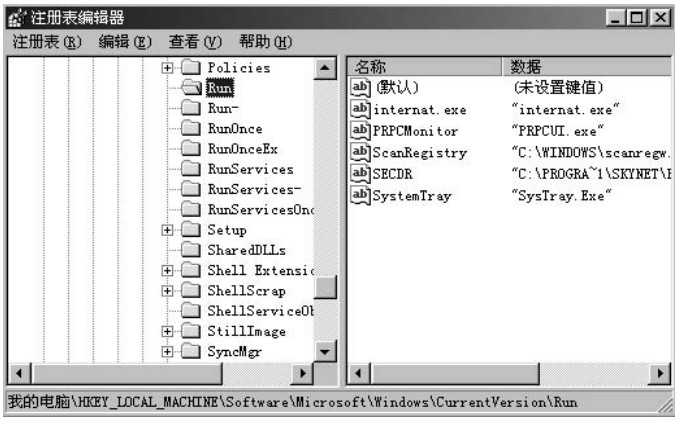


图 1-4

1.4 本章小结

本章介绍了注册表、Windows 及应用程序之间的关系；应用程序与 Windows 整合及注册表的添加项目；注册表项目的修改方法等知识。通过这些知识的介绍，说明了注册表与应用程序编程的关系以及掌握以编程方式实现注册表的修改的重要性。

第二章 注册表结构概述

2.1 注册表的由来



许多读者对于注册表的概念的认识可能是从 Windows 9x 开始的，注册表包含了 Windows 系统和用户程序的各种所需数据及项目配置、硬件信息等许多信息。注册表在 Windows 操作系统中相当重要，也相当神秘。微软公司的操作系统在 Win 3.x 就已经有了注册表的雏形，当时的系统信息主要由 Autoexec.bat、Config.sys、System.ini、Win.ini 和 Reg.dat 等文件构成，其中的 Reg.dat 就是当时的注册表，但是它的功能远不如 Winodws 9x、Windows 2000 等操作系统的注册表强大、稳定。而且微软的视窗操作系统在经历了几代的发展后，注册表的作用变得越来越重要、功能也越来越强大。

2.2 注册表的作用

众所周知，操作系统在用户与计算机之间充当着一个相当重要的桥梁作用。操作系统要接收用户的各种命令并根据这些命令进行硬件资源的调配，如内存分配、内存管理、I/O 管理等。操作系统的出现促进了硬件和软件方面的长足进步，两者相辅相成。操作系统经历了几代发展后，硬件的种类及性能有迅速的提高，操作系统性能及功能已经大大加强，尤其是视窗系统的出现，使得人机交互更加友好。微软的 Windows 操作系统集成了很多应用程序，同时也出现了一些问题：如何更好地管理并利用这些资源，如何使 Windows 应用程序与操作系统更好的整合在一起。

注册表包含了系统的硬件、软件等方面的信息，使得操作系统更好地分配利用硬件资源，同时使得程序对注册表进行简单的修改即可使应用程序与 Windows 操作系统整合在一起。这种整合使得 Windows 可以更好地管理资源及管理进程。在另一层面上，微软的 Windows 操作系统是一个图形界面的比较友好的操作系统，每个用户对工作环境有不同的要求，可以通过修改注册表来达到这种效果，而且这是相当容易的一个过程。因此，从这种意义上来说，即使版本一样 Windows 也有很多种类。

2.3 注册表操作的注意事项

注册表作为 Windows 操作系统最重要的数据库，从内容到结构都是相当复杂的，且注册表本身也只是以文件形式保存，所以注册表又是比较脆弱的。若注册表存在错误或遭到破坏，轻则影响系统性能和稳定，重则不能正常访问硬件或运行软件，甚至无法启动机器或使程序与数据受到损坏。注册表对于 Windows 是非常重要的，对它的微小修改都会对 Windows 操作系统产生影响，甚至会出现系统崩溃等严重的问题。因此对于注册表的任何操作应慎之又慎，建议在进行注册表有关操作前首先要对注册表有个总的了解，切勿盲目操作。下面几项请务必遵守：



1.在修改注册表之前请务必对老注册表进行必要的备分，以便在必要时候可以使用这份备份注册表进行还原。

2.在修改注册表项目之前，读者应保证对此项目有较深刻的理解，或者至少应保证修改有充分的依据。

3.修改前对修改项的原有值进行必要的记录。

同样，编程时也应注意这些事项。由于本书的编程实例均涉及注册表，因此在进行编程调试时，特别需要对注册表进行备份，以后章节的编程实例中将会对程序涉及的注册表项目作出明确的提示，以使读者在进行编程时做到心中有数。

2.4 注册表的文件组成



前面介绍过，Win 3.x 的注册表由 Autoexec.bat、Config.sys、System.ini、Win.ini 和 Reg.dat 等文件构成。Windows 9x 系列及 Windows 2000 的注册表由此发展而来，但它们已经有了本质上的不同。与注册表相关的文件有六个，但最主要的是 System.dat 和 User.dat。这两个文件都具有系统、隐含、只读属性。

1.系统配置的注册表文件 System.dat

System.dat 处于 Windows 目录下，是 Windows 98 注册表的一部分。这个文件的作用很类似于 Win 3.x 中的 System.ini 文件。该文件保存有微机的系统信息，如硬件设备配置和设备驱动程序的有关信息等等，还保存安装即插即用类型的设备硬件配置,如设备的 I/O 地址、IRQ 级和 DMA 通道等的情况。

这个文件在 Windows 98 的网络运行状态时保存在本地的工作站或本地 PC 机系统中。在 Windows 98 的安装期间，安装程序将首先检查你的计算机上已经安装了的硬件设备，然后在 System.dat 中建立适当的配置项。若从现有的 Win 3.x 中升级安装 Windows 98，则安装程序会将 Win 3.x 中的 System.ini 、Reg.dat 文件中的部分设置项拷贝到 System.dat 中，以保持系统的设置。

Windows 操作系统中“控制面板”有一个“系统”图标，单击它可以修改系统的硬件设备和配置情况，此时进行的操作都是从 System.dat 中读取的系统属性设置。

2.用户平台配置的注册表文件 User.dat

User.dat 与 System.dat 一样处于 Windows 系统目录中，该文件的作用类似于 Win 3.x 系统中的 Win.ini 文件，它也是 Windows 98 的注册表的重要组成部分。

这个文件在 Windows 操作系统中同样起着相当重要的作用，保存有用户特征信息，如桌面设置、墙纸或窗口的颜色设置等。它定义用户优先权，如用户平台设置，运行级等等；它特定于某一个用户的应用程序的安装信息，应用程序的使用信息等等。如果你是使用局域网的话，当你在 Windows 98 中使用网络时，User.dat 必须放在网络服务器上。对于单机来说，配置了网络属性的 User.dat 是保存在本机的系统目录下的。

在你第一次输入用户名和密码时，Windows 98 程序将把这些信息存储在 User.dat 中，同时你的 Windows 98 的系列号也存储在 User.dat 中。如果用户在“控制面板”的“密码”图标中选择了“用户可自定义首选项及桌面设置。登录时，Windows 自动启用个人设置”这个选项后，Windows 98 系统将会为每个用户创建他自己的 User.dat，并且把它保存为

C:\Windows\Profiles\用户名\User.dat。在用户每次登录后，他自己的 User.dat 就会被调入到系统中，同时启动自己相应的程序信息。

这两个文件是注册表的主要组成部分，对 Windows 系统极为重要，因此 Windows 操作系统为了保证其稳定性，对这两个文件进行了自备份，分别为 System.da0 和 User.da0。当 System.dat 与 User.dat 遭到破坏时，系统会在启动时由这两个文件恢复注册表。这为不太稳定的 Windows 系列提供了一套保护措施。

除了 System.dat 与 User.dat 这两个文件外，如果在系统中安装了“系统策略编辑器”，还当包括 Config.pol，这个文件的属性同样是系统、隐含、只读，同样保存在 Windows 的系统目录下。Config.pol 主要用于 Windows 98 的网络用户的管理方面的策略，系统以此对网络用户的操作作一些限制。所谓“策略”，就是指系统根据 Config.pol 文件中的设置对网络用户的操作作一些限制。系统里安装了“系统策略编辑器”后，用户可以使用 Config.pol 文件中的限制来决定系统用户的操作权限，如执行程序、修改注册表等等。



同样此文件也有一个自备份文件 Config.po0，此文件一般保存在网络服务器上。

综上所述，与注册表相关的文件主要有六个，三个主文件，三个备份文件，它们分别是：System.dat、User.dat、Config.pol、System.da0、User.da0、Config.po0。值得一提的是，安装完 Windows 后，安装程序会在安装盘的根目录下留下一个名为 System.lst 的文件，此文件实际上是初装完系统后的第一份 System.dat，是最原始的一份。在万不得已的时候可以用它来恢复注册表，但此时已经安装的大部分应用软件将必须重新装过才能使用。

2.5 注册表的组织形式

Windows 操作系统的注册表包含大量的数据，所有这些数据都是通过一种树状结构以键和子键的方式组织起来，就像磁盘文件系统的目录结构一样。通过注册表编辑器 Regedit 可以方便地访问，图 2-1 即为打开后的注册表，读者在编辑器的左窗格中看到六个项目，它们被称为注册表的六个根键，分别为 HKEY_CLASSES_ROOT、HKEY_CURRENT_USER、HKEY_LOCAL_MACHINE、HKEY_USERS、HKEY_CURRENT_CONFIG、HKEY_DYN_DATA。这些根键的作用如下：

1.HKEY_CLASSES_ROOT

管理文件系统。根据在 Windows 98 中安装的应用程序的扩展名,该根键指明其文件类型的名称，打开该文件所要调用的程序等等信息。

2.HKEY_CURRENT_USER

管理系统当前的用户信息。在这个根键中保存了本地计算机中存放的当前登录的用户信息,包括用户登录用户名和暂存的密码。在用户登录 Windows 98 时，其信息从 HKEY_USERS 中相应的项拷贝到 HKEY_CURRENT_USER 中。

3.HKEY_LOCAL_MACHINE

管理当前系统硬件配置。在这个根键中保存了本地计算机硬件配置数据，此根键下的子关键字包括在 System.dat 中,用来提供 HKEY_LOCAL_MACHINE 所需的信息，或者包括在远程计算机中可访问的一组键中。这个根键里面的许多子键与 System.ini 文件中设置项类似。

4.HKEY_USERS

管理系统的用户信息。在这个根键中保存了存放在本地计算机口令列表中的用户标识和密码列表。同时每个用户的预配置信息都存储在 HKEY_USERS 根键中。HKEY_USERS 是远程计算机中访问的根键之一。

5.HKEY_CURRENT_CONFIG

管理当前用户的系统配置。在这个根键中保存着定义当前用户桌面配置(如显示器等)的数据,该用户使用过的文档列表（MRU），应用程序配置和其他有关当前用户的 Windows 98 中文版的安装的信息。

6.HKEY_DYN_DATA

管理系统运行数据。在这个根键中保存了系统在运行时的动态数据，此数据在每次显示时都是变化的，因此，此根键下的信息没有放在注册表中。

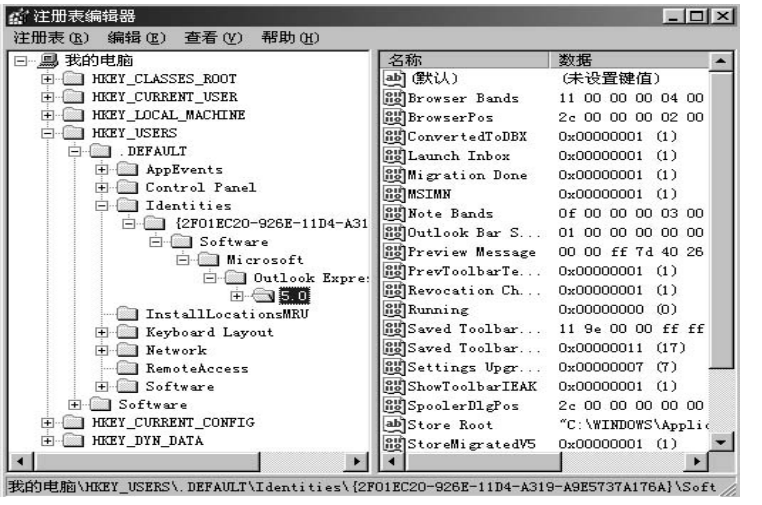


图 2-1

每个键都包含了一组特定的信息，每个键的键名都是和它所包含的信息相关联的。从图 2-1 中可以看出，如果某个键包含了子键，则在注册表编辑器窗口中代表这个键的文件夹的左边将有“＋”符号，以表示在这个文件夹中有更多的内容。用鼠标点击“＋”号可以打开文件夹，打开后的文件夹前面的“＋”就会变成“－”，这样用户就可以一层层地开注册表树。注册表编辑器还提供了另一种打开方式，即采用查找方式。当用户不清楚要找的键在哪个目录分支下面，就得搜索相应的关键字。

2.6 数据类型

2.6.1 各根键之间的相互关系

六个根键由大写字母和下划线组成，并都以 HKEY_为前缀，每个根键下都有数量众多的子键和键值。尽管有六个根键，但这六个根键并非完全独立、毫无关系，它们之间存在着某种联系，尤其是在以下几个方面：

1.HKEY_CLASSES_ROOT 根键的全部内容包含在 HKEY_LOCAL_MACHINE 中，它的全部内容处于 HKEY_LOCAL_MACHINE\SOFTWARE\Classes 下。

2.HKEY_CURRENT_CONFIG 中的内容也可以在 HKEY_LOCAL_MACHINE 中找到，它的全部内容处于 HKEY_LOCAL _MACHINE＼Config 下。

3.HKEY_CURRENT_USER 仅是 HKEY_USER 中的一部份信息。Windows 将会根据系统启动时登陆的用户不同，而选择将 HKEY_USER 中的某部分信息复制到 HKEY_CURRENT_USER 中。

2.6.2 注册表键值项数据类型



在注册表的各级键与子键之下保存有大量的信息，这些信息都是以各种形式的键值项数据来保存的。各键值项数据显示在注册表编辑器的右窗口中。各项数据分为键值名（名称）和键值（数据）两部分。键值名是各键值项数据的标识符，而键值就是各键值项数据的具体内容了。键值分为如下三类：

1.字符串值。它一般用于描述文件或各类标识。通常由字母和数字组成，最大长度为 255 字符。

2.二进制值。它的长度不受限制，在注册表编辑器中以十六进制的方式显示，如“60 60 00 00”。

3.双字（DWORD）值。它是一个长度为 4 个字节的数值，它也是以十六进制的方式显示的。数值前加“0x”与二进制区别，形如“0x00000000（1）”。

2.6.3 注册表数据类型

注册表的键中包含着各种不同格式的数据。数据类型可以这样来分类：

1.通用数据类型：RegEdit、RegEdt32 及其它绝大多数注册表工具都支持，并能够对之进行编辑的数据类型。

2.Windows NT 专用数据类型：RegEdt32 和另外几个注册表工具支持，并能够对其进行编辑的数据类型。

3.组件／应用程序专用的特殊数据类型：注册表工具支持这些数据类型，但是对于程序而言是有限度的，用户则只能将其作为二进制数进行编辑。

实际上，注册表工具也可以对不支持的数据类型进行编辑，包括那些显示为 REG_UNKNOWN 类型的数据。但是编辑仅能在二进制模式下进行，这就需要用户对数据对象的格式非常了解。对于那些需要手工修改注册表的用户来说，理解每一种数据类型以及每一种类型数据的存储格式是非常重要的。表 2-1 列出了现有的注册表数据类型。

表 2-1 现有的注册表数据类型

类　型	索引	大　小	说　明
REG_BINARY	3	0~多个字节	可以包含任何数据的二进制对象颜色描述
REG_DWORD	4	4 字节	DWORD 值
REG_DWORD_BIG_ENDIAN	5	4 字节	一个 DWORD 值的逆序存储形式
REG_DWORD_LITTLE_ENDIAN	4	4 字节	DWORD 值
REG_EXPAND_SZ	2	0~多个字节	包含环境变量占位符的字符串

需要全本请访问 www.ertongbook.com