打开 Windows 这扇窗

深入 WINDOWS 编程

Windows 加密及压缩软件编程技巧与方法

雷军　王全国　马贤亮　著

清华大学出版社

打开 Windows 这扇窗

深入 Windows 编程

——Windows 加密及压缩软件编程技巧与方法

雷　军　王全国　马贤亮　著

清 华 大 学 出 版 社

(京)新登字 158 号

内 容 提 要

本书主要讲述 Windows 核心运行机制与通用 Windows 加密软件和压缩软件编程技巧与方法。本书以 Windows 的可执行文件格式为基本点进行全方位展开，介绍了众多的 Windows 分析工具并给出了完整的源代码，以清晰的逻辑关系展现了复杂的 Windows 执行机制以及 Windows 程序自装载技术，还有如何直接修改 Windows 执行文件、如何用汇编语言编写 Windows 应用程序等内容，解决了编写通用 Windows 加密软件的技术难题，本书最后介绍了 Windows NT 和 Chicago 的可执行文件格式及分析工具。随书赠送的磁盘中有书中介绍的示范程序的源程序和一些工具软件，有助于读者透彻地理解 Windows 的核心。

本书深入浅出，适合于所有对 Windows 有兴趣、略懂 DOS 编程技术的读者。

好的工具可以让你事半功倍；同样的，一本好书，可以帮助你在知识的领域里更上一层楼。

图书在版编目(CIP)数据

深入 Windows 编程：Windows 加密及压缩软件编程技巧与方法/雷军等著. —北京：清华大学出版社，1994.12

ISBN 7-302-01673-9

Ⅰ.深… Ⅱ.雷… Ⅲ.操作系统(软件)-程序设计 Ⅳ.TP316

中国版本图书馆 CIP 数据核字(94)第 13730 号

出 版 者：清华大学出版社(北京清华大学校内，邮编 100084)
责任编辑：蔡鸿程
印 刷 者：清华大学印刷厂
发 行 者：新华书店总店北京科技发行所
开　　本：787×1092 1/16　印张：24.25　字数：601 千字
版　　次：1994 年 12 月第 1 版　1994 年 12 月第 1 次印刷
书　　号：ISBN 7-302-01673-9/TP·719
印　　数：0001—8000
定　　价：59.00 元

致　谢

我们的力量来源于众多朋友的支持和鼓励，在此谨向他们表示诚挚谢意。

香港金山公司副总裁求伯君先生的大力支持，同事傅占华、胡卫翔、陈波、冯志宏、钱国祥等不遗余力的协助，才使本书在北京金山软件公司的机房里顺利完稿。美国 Aldus 公司的张湘辉提供了一些资料，Rosenthal Engineering 的 Doren Rosenthal 试用过 PACKWIN，这两位先生给了我们相当大的帮助。

雷　军： 心中涌动着无法言语的感激，尤其是对：带我入门的武汉大学计算机系张德向等老师，最早在自己的软件产品中选用 BITLOK 的联邦软件产业发展公司总裁苏启强，给我们提供了一个创造和表现舞台的香港金山公司副总裁求伯君，还有一起奋斗的弟兄们。

王全国： 尽管我的妻子和女儿对 Windows 并不关心，但我还是要把这本书献给她们，因为她们使我明白：Windows 外面的世界更美好！

马贤亮： 献给飞扬、迷茫的青春感觉。

目 录

引言

中国软件走向世界

0.1 令人振奋的机遇与紧迫的挑战

自从 Microsoft 于 1990 年 5 月推出第一个 Windows 的成熟版本 Windows 3.0 以来，整个计算机工业界围绕 Windows 系列（包括 Windows 3.1、Windows NT、Chicago 和 Cario 等等）进行了大规模的分化和重新组合。目前，全球有 5000 万 Windows 用户，Windows 上的应用软件已逾万套。这一汹涌澎湃的 Windows 潮流给了我们一个全面赶超世界软件水平的机遇。这是因为：

Windows 与 DOS 用户界面完全不同，直接以图形模式为基础。Windows 提供与设备无关的图形操作，比如画线、画矩形和圆以及其它更复杂的区域。由于是设备无关的，所以同一功能可以在不同的打印机和监视器上工作。例如，你不用告诉程序所联的是什么打印机，只需简单地调用 Print 命令即可。Windows 的设备驱动程序将这些图形操作转换输出到打印机、显示屏或其它的输出设备。Windows 的这种图形模式，使 Windows 环境真正摆脱了对硬件文本功能（如显示卡、打印机）的依赖；同时，Windows 中各种外设均以 Driver 形式嵌入系统，对系统而言，面对的是由 Driver 描述的“虚拟设备”。Windows 的这种设备无关性要求各种设备根据 Windows 的驱动程序接口，重新编写不同于 DOS 的驱动程序，使之能够输出各种图形和文字。所以说，Windows 的这种基于图形模式的外设无关性为多文种支持（包括中文）奠定了基础。

其次，在系统职能上，Windows 相对于 DOS 有很大的增强。Windows 不仅增加了作为一个操作系统所必须的内存管理、进程管理等基本部分，吸收了多媒体、网络、笔式输入等新技术，还扩充了资源、字体、文字处理、对话、编辑等原本由应用软件处理的功能。大量的功能由应用级转移到系统级，既保证了风格的统一，简化了应用软件的开发，又使应用软件的注意力保持在问题分析和解决方案上（What to do），而不是在方案的实现上（How to do）。上述功能的实现与具体的应用软件无关，这是软件兼容性的重要保证，也是 Windows 作为一个新的软件平台流行的原因。

目前的 Windows 就是一个包含初步国际化功能的产品。它不仅在时间格式、货币符号和键盘分布等方面提供了国际化支持，使用 8 位的 ANSI 字符集，还保留了对双字节字符（DBCS）内码的支持能力，只是在 Windows 的西文版本中没有实现而已。此外，Windows 提倡把提示信息（如菜单、对话、字符串等）都放在可执行程序的资源段中的风格，有利于不同

语言版本的转换。目前，中文之星(新天地)、PWin(Microsoft)、金山皓月(香港金山)、Win-MATE(四通利方)等 Windows 中文环境已陆续出现。它们以中西文兼容为突破口，消除了中西文软件应用与开发的技术障碍，基本构成了实用开放的中文系统。今后的中文平台将朝着国际化的软件操作环境、完善的中文支持(包括软件中的文化因素)、任意的软硬件适配等方向发展。即将出现的下一个 Windows 版本 Chicago 能够直接处理中文，任何应用软件都不需要“中文本地化”(汉化)。Windows 国际化的特点使得语言不再是阻碍我国软件开发的主要因素。

再者，Windows 作为一个崭新的操作系统，尽管国外的开发者可以早一步拿到完整的资料、拥有众多的软件，但从总体上讲，国内和国外的开发者仍处于同一起跑线，大家都需要一段时间才能熟悉 Windows，国内外的软件同行在 Windows 上的差距是很小的。如我们开发的 Windows 执行文件压缩软件 PACKWIN，和国外同期开发的类似产品相比，效果要好得多。这说明在一些具体的应用软件开发上，我们还是有可能走在国外软件开发者前面。

综上所述，Windows 跨越语言的差异，作为一个理想的软件开发平台，它使中国的用户有可能同步享受世界最新的软件精品，更使中国的软件工程师得到了一个与世界各国同行共同开拓的契机。

Windows 为我们提供了一个舞台，但这是一个需要奋力搏击的舞台。Windows 给了中国软件产业一次机会，同时也把中国的软件市场推向了世界。因为 Windows 的开放性，使国外软件更容易突破中文处理技术的壁垒，于是，国外的优秀软件扬起 Windows 风帆，纷纷来到中国：Word, Excel, FoxPro, Lotus 1-2-3, AmiPro, Word Perfect...

这是对民族软件工业的一次挑战。面对这一个相互竞争的市场，软件开发人员应该尽快进入 Windows 世界，积极吸收国外先进的技术；同时，在实践中逐步培养“软件工程”的观念和经验，把软件开发真正作为一个工程，开发出世界级顶尖产品。

虽然我国潜在的市场很大，但目前相对国际市场来说还是很小的，估计还不到 1%。我国有着世界优秀的软件工程师，应该好好把握 Windows 这一次新的机会与挑战，创造世界名牌，让中国软件走向世界。

打开 Windows 这扇窗，外面的世界更精彩！

0.2 编写目的

两年前，香港金山公司的 WPS 文字处理系统就已风靡全国，成为装机量仅次于 DOS 的软件产品。为了拓展产品范畴，公司决定由北京金山软件公司开发 Windows 上的产品“双城电子表”。那时，我们对 Windows 的了解比较肤浅，对 Windows 核心还一无所知。

因为“双城电子表”不是一、二万行就能解决的小项目，随着程序量的增大，在调试中我们经常碰到一些莫名其妙的问题，而在参考手册中找不到这些问题的解决办法；再者，我们开发的“双城电子表”及一系列 Windows 上的应用产品都即将面市，必须解决加密问题；同时，我们又不安心于总在“Windows 黑匣子”外面逗留。实际需求加上内心的冲动，迫使我们逐步深入到 Windows 底层中去。这本书奉献给大家的就是我们这一方面两三年经验的积累。

0.3　为什么要深入 Windows 核心

Windows 是什么？对于一般的用户而言，它是菜单、按钮、对话框、多个窗口、能够同时执行的多个程序，是一个比 DOS 强大的运行环境。对于编程人员而言，首先，Windows 是一套完备的图形界面接口，编程人员只需按照 Microsoft Windows 文档中的规定传递参数，调用函数，而不用知道更进一步的细节。比如说，编程时经常要用到各种 handle(selector)，它们分别指向相应的数据结构，而编程人员并不需要知道这个数据结构的具体内容 — Windows 把这些信息隐藏了，这也正符合结构化程序设计中所强调的“信息隐藏”观念；其次，Windows 是许多生硬的定义的组合，Windows 实现了针对这些定义的操作，但这些被隐藏在背后。对 Windows 开发人员——如果他有刨根问底的兴趣的话——这些操作的实现细节象一个魔术袋，开发人员只能站在外围看魔术师表演，却没有机会把头探进魔术袋看个究竟，当然也就无法揭穿魔术师的把戏了！这种血肉分离的定义是生硬的。比如，Microsoft 的文档中给出了 NE 文件结构中每一项的说明，与 NE 文件结构相关的基本操作是：Windows 把一个磁盘上的 AP 或 DLL 运行起来需要经过载入、启动、执行等步骤，中间还涉及动态连接等其它方面，但 Windows 把这些事情全部接管了，编程人员从 Microsoft 的文档中无法找到 Windows 执行机制的细致描述，也就难以编写自己的装载和启动程序。Windows 实施这种策略的后果是：程序员对 NE 结构只有静态的感觉，没有与执行机制融合在一起的生动理解。

现在的问题是，我们必须超越“Windows 高级编程”，深入到 Windows 核心，从高处把握 Windows。比如说，我们需要知道 Module Handle、Task Handle、Instance Handle 之间的关系；还要为自己的 Windows 压缩、加密程序编写自装载代码；我们还想与 Windows 平起平坐，共同操纵执行程序以满足特别的需要，而不愿意把事情全权委托给 Windows，让它用自己缺省的方式在“暗室”里处理完毕后再交回来……

那么，如何深入到 Windows 核心呢？我们选择了“Windows 可执行文件”为出发点。因为一个操作系统的可执行文件格式从多种意义上讲就是该系统本身执行机制的反映，而且研究可执行文件格式能够积累大量的知识。本书的主要内容正是围绕 Windows 可执行文件格式展开的。

0.4　本书的结构

● **第 1 章　分析 Windows 执行文件**

本章引用了 Microsoft 的参考手册中关于 NE 文件格式的说明，并修正了原文中的一些错误，添加了一些保留或未予说明的内容。NE 文件格式的说明包括 WINSTUB，NE 信息块，段表、资源表、驻留名表、模块引用表、输入名表、入口表、非驻留名表，代码段和数据段的重定位信息，以及几种主要资源的结构。本章对以上内容都使用二

进制 dump 方式给予详细讲解，以便读者对 NE 结构有形象的理解和直观的记忆。

● **第 2 章 Windows 执行文件的分析工具**

如果你已经了解了 NE 结构，自然不会再喜欢用原始的 dump 方式去察看文件。市面上有许多流行的 NE 文件分析工具，如 TDUMP、EXEHDR、MAPWIN、EXEDUMP、NEWEXE 等，它们都能将 NE 文件中的执行信息以简洁明了的文本形式给出。这些工具有各自的侧重点，代表了不同的需要。本章的重点是介绍作者研制的 Power 系列分析工具中的两个：Power Dump 和 Power FileInfo，并通过用上述工具分析实例来强化第 1 章中讲述的概念。

● **第 3 章 分析工具的开发实例**

探索 Windows 的第一步是深入到 Windows 可执行文件中去。在前面两章中，我们对 NE 格式的了解只是概念性的。为了熟练驾驭结构复杂的 Windows 可执行文件，本章首先定义了一个文件对象 File Object，接着向读者演示如何开发一个类似于 EXEHDR 的分析工具 MSDUMP。选择这一个演示实例是基于以下的两个理由：首先，知道 NE 中的执行信息的含义和能够编写程序获取这些执行信息是不同的两回事 — “概念上理解”和“实际编程”是不同的两回事；再者，任何针对 NE 文件的操作都必须以正确地读取 NE 中的信息块、各种表以及重定位信息为基础，MSDUMP 这一个例子正好适合。在演示 MSDUMP 的开发过程之前，本章中还介绍了一个扩展工具集 ExtTools，该工具汇集了编程中经常要用到的一些操作，因而它使应用程序的主要意图清晰明了。

● **第 4 章 直接修改 Windows 执行文件**

如果没有一个执行文件的源程序，我们只有硬着头皮去直接修改。进行这种操作需要实施者对 Windows 执行机制有全面的了解。所谓执行机制，通俗地讲，就是 Windows 系统把一个执行程序从磁盘上载入到内存中，并根据 NE 信息块和文件头中各种表格的内容，在内存中为该程序维护相应的数据记录，初始化与该程序有关的堆栈、事件队列等，从重定位表中的信息准确地找到重定位的代码，以及在程序执行时处理各种问题的原则和方法的总称。本章的重点是叙述与执行机制相关的概念、数据结构以及他们之间错综复杂的动态关系。以这些知识为基点，本章演示了直接修改 Windows 执行文件中某些部分的方法，读者可以参照这个示范修改其它 NE 程序。

● **第 5 章 用汇编语言编写 Windows 应用程序**

虽然 C 语言是编写 Windows 程序的最佳语言，但至少有以下两个理由说明在有些场合还必须用汇编语言编写 Windows 程序：一是涉及到一些低级操作，如针对 BIOS 的读写；另一种情况是在那些必须追求效率的场合，如编写压缩软件。然而，即使你对 DOS 环境下的各种汇编语言相当熟练，编写 Windows 汇编程序并非一件易如反掌的小事：Windows 汇编引入了一些新的宏定义，它有自己的编程约定！

在这一章中，我们强调“用汇编语言编写 Windows 的方法”是出于以下两个目的：一是在本书第 6、8、9 章中有多个用汇编语言编写的示范程序；另外一点是，我们认为，一个能够用汇编语言编写 Windows 程序的程序员，他阅读、跟踪、调试实际运行代码不会遇到什么大的困难，同时，他又具备了深入 Windows 核心的关键手段：因为

只有汇编语言才是与机器语言直接等价的。

● **第 6 章　自装载 Windows 执行文件**

在可执行文件启动和执行以前，存在一个装载过程。对于符合 NE 格式标准的文件，装载过程一般是交由 Windows KERNEL 按缺省方式自动完成。但是，如果 NE 程序被加密或压缩过，它的执行信息就被破坏了，那么在装载时就需要首先把被变形的程序还原成标准的格式；如果程序的某个段大小超过了 64K，它就不再是符合标准的格式，Windows 不能接受，因而装载程序必须为这样的段分配内存。本章讲解了自装载 Windows 程序时用到的装载函数、数据表，并以 HELLOWIN 为例，给出了一个完整的自装载实例。自装载技术是操纵 Windows EXE 时最关键、最复杂的技术，也是笔者分析 Windows 的最大收获。

● **第 7 章　LZ 压缩算法原理与具体实现**

Windows 可执行文件通常占用很大的存储空间，良好的数据压缩技术可以有效地利用有限的存储介质。本书第 8 章将实现一个 Windows EXE 压缩软件 PACKWIN。这一章向不熟悉压缩技术的读者讲述了数据压缩概论、LZ 压缩算法原理、压缩与还原算法的实现等基础知识，最后给出了 PACKWIN 中用到的压缩函数的实现细节。

● **第 8 章　开发 Windows 执行文件压缩软件**

这里所指的“压缩软件”要求能够随着 Windows EXE 的执行，自动解压缩相关的段和资源，使之符合标准的 NE 格式。本章向读者介绍了 PACKWIN 是如何压缩 STUB，如何压缩代码段、数据段及 Windows 资源；如何在被压缩的文件中插入自装载段，又如何在自装载段中的 LoadAppSeg 函数里实现解压缩。PACKWIN 实现涉及到了第 4 章“直接修改 Windows 执行文件”、第 6 章“自装载 Windows 应用程序”、第 7 章“LZ 压缩算法原理与具体实现”以及第 3 章中的 File Object 定义，它汇集了操纵 Windows EXE 时经常使用的手段，所以“PACKWIN”是一个比较典型的主题。

● **第 9 章　开发 Windows 加密软件**

Windows 取代 DOS 成为微机上新的软件开发平台已是事实，很多的软件厂商都在开发基于 Windows 的应用软件。在软件的法律保护并不完善的今天，过去常用的通过加密手段来保护软件版权的办法，同样也遇到了一个从 DOS 到 Windows 转变的问题，这种转变的主要困难在于 NE 新格式和 Windows 特殊的执行机制等方面。BITLOK for Windows 在 BITLOK 的基础上，针对 NE 的特点，成功地实现了对 Windows EXE 的加密。本章讲述了软件加密的一般知识，并给出了 BITLOK for Windows 的实现方法。

● **第 10 章　Windows NT 及 Chicago 执行文件格式**

在不久的将来，Windows NT、Chicago 等系统将逐渐取代 Windows 3.1。Microsoft 为这些基于 Win32 的系统设计了一种不同于 NE 的新文件格式 — PE 格式。因为系统可执行文件格式最能反应这个操作系统本身的特性，所以我们仍然选择 Win32 执行文件格式来引导读者深入到这些未来的操作系统内部。本章较详细地介绍了 PE 格式。鉴于本书前面对 16 位 Windows 的描述，我们在介绍基于 Win32 的 PE 文件格式时，将它和 16 位 NE 文件格式作了比较对照。本章还将第 2 章中介绍过的 PDUMP 加以改造，使之能够分析 PE 格式的文件。

0.5 如何使用本书

如果你仅仅对于 Windows 有些兴趣，只需要翻翻即可。需要的时候，再详读重点章节。

如果你需要编写 Windows 下的压缩或加密软件，必须配合磁盘中的实例，系统地阅读本书的每个章节。

不管你是 DOS 资深专家还是 Windows 初学者，阅读本书都不会遇到什么困难。

0.6 磁盘资料

本书所附的软盘中，不仅包含了各章节中示范程序的源代码，同时还有许多可以立即使用的工具程序。盘片中的具体资料说明如下，如有修改，以盘中文件为准，不另外说明。

README	磁盘文件的说明
TPU.ZIP	编译 Pascal 程序所用到的 TPU
PKUNZIP.EXE	展包程序
\Chap1	
WINSTUB.EXE	一般的 STUB 程序
LOADSTUB.EXE	在 DOS 下能够装入 Windows 的 STUB 程序
MINISTUB.EXE	最小的 STUB 程序
MINI.BAT	生成 MINISTUB 的批文件
MINISTUB.ASM	MINISTUB 的汇编源程序
COM2EXE.EXE	COM 文件转换成 EXE 文件
COM2EXE.PAS	COM2EXE 的源程序
RESTUB.EXE	修改 STUB 的工具
GETSEG.EXE	取某段的工具
GETSEG.PAS	GETSEG 的源程序
PACKBMP.EXE	压缩 BITMAP 的工具
PACKBMP.PAS	PACKBMP 的源程序
\Chap2	
TOOLS.ZIP	包含许多著名的 Windows 分析工具
PDUMP.EXE	黄玫瑰开发的分析文件格式的工具
PDUMP.PAS	PDUMP 的源程序
PFI.EXE	黄玫瑰开发的分析文件格工的工具
PFI.PAS	PFI 的源程序
\Chap3	
EXTTOOLS.PAS	EXTTOOLS 的接口说明部分
FILEDEF.PAS	File Object 的接口说明
MSDUMP.EXE	利用 File Object 开发的类似 EXEHDR 的工具
MSDUMP.PAS	MSDUMP 的源程序
\Chap4	
ADDCODE.EXE	附加一段代码的工具
ADDCODE.PAS	ADDCODE 的源程序
RESTUB.PAS	修改 STUB 工具的源程序

\ Chap5

HELLOWIN.ASM	HELLOWIN 的 MASM 版
HELLOWIN.DEF	HELLOWIN 的定义文件
HELLOWIN.LNK	链接 HELLOWIN 的定义文件
MAKEFILE	MAKE 文件
HELLOWIN.EXE	

\ Chap6

APPLOAD.ASM	自装载的源程序
RELOC.ASM	Windows 重定位的源程序
HELLOWIN.C	开发一个带自装载功能的 HELLOWIN
HELLOWIN.DEF	HELLOWIN 的定义文件
MAKEFILE	MAKE 文件
C.BAT	编译批处理程序
HELLOWIN.EXE	带自装载功能的 HELLOWIN

\ Chap7

LZPACK.TPU	压缩程序库
LZPACK.PAS	压缩库的源程序
ZLITE.EXE	DOS 版执行程序压缩工具
ZLITE.PAS	ZLITE 的源程序

\ Chap8

PACKWIN.EXE	DOS 和 Windows 执行文件压缩工具
PACKWIN.PAS	PACKWIN 源程序

\ Chap9

BLW.EXE	BITLOK for Windows 1.0 加密主程序
BLWINST.EXE	BITLOK for Windows 1.0 硬盘安装程序

\ TOOLS

RI.COM	RAMINIT DOS 的内存清理工具
RI.DOC	RI 的使用说明
RI.ASM	RI 的源程序

0.7　关于编程风格

一个优秀的程序员必须具备良好的编程风格，一个编程风格好的程序能极大简化调试和维护。本书提供了大量用 Turbo Pascal 编写的示范程序，如第 3 章中的 ExtTools.pas 和 FileDef.pas 等。希望读者仔细研读这些程序，提高自己的编程能力和技巧。本书选用 Borland Pascal 作为主要的示范编程语言，是因为 Pascal 描述算法比较清楚，习惯使用其它语言的读者能够很轻松地看懂或者将示范程序转换成其它语言。

0.8　神秘“黄玫瑰”

一个满脸胡须、经常嗑着葵花籽的人物，他有一把十几个枪管的手枪。此人来无影、去无踪，枪法奇准。他以勇猛、果断的方式除暴安良、杀富济贫。每次“革命”行动之后，总会留下一朵黄玫瑰再飘然离去……。这是罗马尼亚电影《神秘的黄玫瑰》的