HZ BOOKS

微软公司核心技术书库

Microsoft

Windows

附 赠
CD-ROM

Windows 2000
内部揭密

Inside Microsoft Windows 2000, Third Edition

(美) David A. Solomon
Mark E. Russinovich 著

詹剑锋 张文耀 黄 艳 等译

机械工业出版社
China Machine Press

微软公司核心技术书库

Windows 2000 内部揭密

（美） David A. Solomon
Mark E. Russinovich 著

詹剑锋 张文耀 黄 艳 等译

机械工业出版社
China Machine Press

本书深入揭示 Windows 2000 内部结构和运行机制，涉及 Windows 2000 最基础的系统组件和基本概念。主要内容包括系统体系结构、系统机制、管理机制、内存管理、安全机制、I/O 系统、文件系统、网络体系等。本书用大量实验展示了 Windows 2000 的内核，有效地使读者深刻地理解 Windows 2000 系统，充分利用该系统进行应用开发。配套光盘包含本书电子版，以及展示 Windows 2000 内核的工具。

本书版权登记号：图字：01－2000－3118

图书在版编目（CIP）数据

Windows 2000 内部揭密/（美）所罗门（Solomon, D.A.），（美）罗森欧维斯（Russinovich, M.E.）著；詹剑锋等译. －北京：机械工业出版社，2001.10

（微软公司核心技术书库）

书名原文：Inside Microsoft Windows 2000, Third Edition

ISBN 7－111－09100－0

Ⅰ.W... Ⅱ.①所...②罗...③詹... Ⅲ.窗口软件，Windows 2000 Ⅳ.TP316.7

中国版本图书馆 CIP 数据核字（2001）第 045663 号

机械工业出版社（北京市西城区百万庄大街 22 号 邮政编码 100037）

责任编辑：宋 宏 张鸿斌

北京昌平奔腾印刷厂印刷 ·新华书店北京发行所发行

2001 年 10 月第 1 版第 1 次印刷

787mm×1092mm1/16·36.5 印张

印数：0 001-5 000 册

定价：69.00 元（附光盘）

序　言

我很感谢作者给我这个机会为这么重要的书写序言。

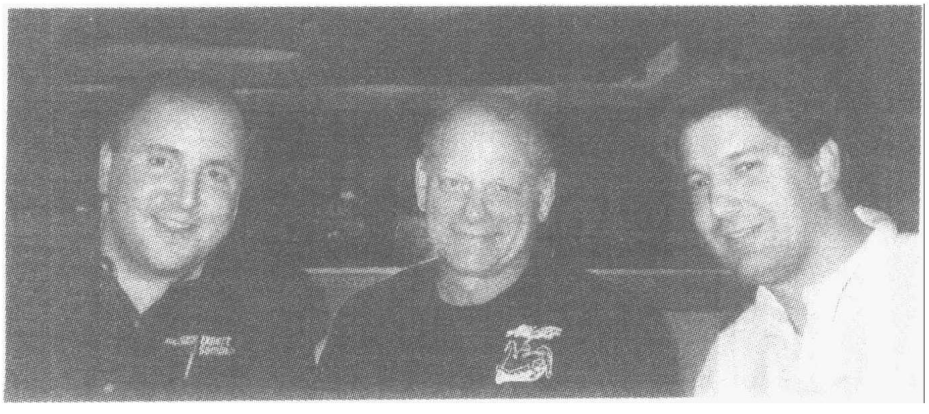
当我在 Digital Equipment Corporation（数字设备公司）工作时，第一次结识当时只有 16 岁的 David Solomon，当时我在为 VAX 开发 VMS 操作系统。从那时开始，他涉及操作系统的开发和操作系统内部的讲解工作。我与 Mark Russinovich 经常接触是最近的事，但是我很早就知道他对操作系统有很深的见解。他做了很多惊人的工作，例如在 Microsoft Windows 98 上运行的 NTFS 文件系统和动态 Microsoft Windows 2000 内核调试器，该调试器可以用来研究正在运行的 Windows 2000 系统。

Microsoft Windows NT 系统开发于 1988 年 10 月，目的是创建一种解决 OS/2 兼容性、安全、POSIX、多处理、网络集成和可靠的可移植系统。随着 Windows 3.0 的出现和巨大成功，开发这个系统的目的变为与 Windows 直接兼容，并且把 OS/2 兼容系统变为子系统。

我们最初认为可以在两年内开发出 Windows NT 系统，但结果直到四年半后的 1993 年夏天 Windows NT 才发布。这一版本支持 Intel i386、ntel i486 和 MIPS R4000 处理器。六个星期后我们又推出了支持 Digital Alpha 处理器的版本。

Windows NT 第一版比预期的规模大，时间晚，所以下一个推出的计划叫 Daytona，这是以佛罗里达州的高速公路命名的。这个版本的目的是减小系统的占用空间，提高系统速度，当然也使它更加可靠。在 1994 年秋天推出 Windows NT 3.5 的六个月之后，我们又推出了 Windows NT 3.51，这是最新的可以支持 IBM PowerPC 处理器的版本。

Windows NT 下一个版本的推出是为了更新用户界面以便与 Windows 95 相一致，并且集成在微软已开发了几年的 Cairo 技术。这个版本用了两年多的时间，于 1996 年夏推出，这个版本就是 Windows NT 4.0。



左起：David Solomon、David Cutler、Mark Russinovich

接着推出了 Windows 2000 系统。同以前的版本一样，Windows 2000 也是基于 Windows NT 技

术建立的，并且增加了像活动目录（Active Directory）这样的新的重要特性。开发 Windows 2000 用了三年半时间，它是经过多次测试和改版的 Windows NT 的最新版。Windows 2000 是对四个体系结构经过 11 年开发所达到的顶峰。它的代码库正在向新的 Intel IA－64 体系结构移植。Windows 2000 是目前我们开发的 Windows NT 技术的最好版本，我们还将继续努力开发新的版本。

本书是介绍 Windows 2000 的内部结构和运行机制的权威著作。作者在 Windows NT 的代码细节上曾经做过大量工作，在本书中使用了很多工具并例举了大量实验以帮助读者理解该系统的工作原理。本书应成为系统开发人员的必备参考书。

微软公司高级突出贡献工程师

David N. Cutler

艾 奇 序

1996 年 8 月我们开始开发 Microsoft Windows 2000。大约三年半之后即 1999 年 12 月 15 日，我们开发了 Windows 2000 Professional、Windows 2000 Server 和 Windows 2000 Advanced Server，并投入生产。在 5000 名工作人员的共同努力下，Windows 2000 成为微软乃至全电脑业中最大的操作系统，它也是我们所生产的最可靠、最综合的系统。

现在，世界上许多互联网站和大型企业都在使用 Windows 2000，Windows 2000 开始成为商业乃至家庭中的标准服务操作系统。Windows 2000 包括许多使我们惊奇的新技术，它可以用于桌面机或笔记本电脑，并且有许多服务功能，包括文件、打印、Web、数据库、事务处理、拨号、路由、流媒体、业务流程应用等等。了解所有这些功能需要花费很多精力，但如果你从这个系统的内核概念开始理解，则更容易掌握这些功能。

如果你像我一样想知道系统究竟是怎样工作的，那么仅仅阅读指导手册或帮助文件是远远不够的。本书将介绍 Windows 2000 内部工作机制，介绍如何更好地利用它，如何最大限度地确保运行的可靠性和安全性，如何判断错误的根源等等。

本书在展示 Windows 2000 的内部技术上做了很多工作。本书包含了直接用于实验和进行诊断的工具，这是本书重要的资源。读完本书后，你会更深入地理解这个系统是如何协调的，新版本是如何改进的，并且会更充分地利用它。

虽然我对 Windows 2000 很熟悉，但读了本书后，我知道了许多以前不了解的东西。打开了本书便如同揭开了有史以来最重要的操作系统的面纱。

微软公司平台部副总监

Jim Allchin

前　言

本书是供计算机专业人员使用的，主要面向开发人员和系统管理人员。它介绍了 Microsoft Windows 2000 操作系统内部的工作原理。本书可帮助开发人员在 Windows 2000 平台上开发特殊功能时更好地理解设计方案的基本原理，也可以帮助开发人员解决复杂的难题，还可以帮助系统管理员理解系统运行的规律并解决系统故障。阅读本书可以深入了解 Windows 2000 的工作原理和运行规律。

本书的结构

本书前两章介绍了全书所用的基本术语和概念。接下来的三章讲述了系统的基本工作机制，即分别介绍了系统机制、启动与关机、系统管理机制。后面章节分别介绍了进程、线程和作业，内存管理，安全机制，输入输出系统，存储管理，高速缓存，文件系统和网络。几乎涉及了所有 Windows 2000 操作系统的内核概念。

第三版的特征

本书是第三版，包括了许多《Inside Windows NT》第二版没有的内容，例如启动与关机、内部服务、内部注册、文件系统驱动和网络。包括 Windows 2000 改进和增强，例如 Windows Driver Model（WDM）、即插即用、电源管理、Windows 管理装置、加密、作业对象和终端服务。

本书也是第一次附带一张光盘，包括了浏览 Windows 2000 系统内部的工具。利用光盘也可以查阅本书的电子版。另外书中增加了许多实验，例如讲述利用内核调试器观察 Windows 2000 系统内部状态。

实验

当涉及到某个工具可以用来揭示或演示 Windows 2000 内部运行的特征时，便在“实验”框里介绍其步骤。本书中有很多这样的例子。你一边阅读就可以一边看到 Windows 2000 内部工作的过程。这样比只是看书印象要深得多。很多实验都要用到内核调试器。本书配套光盘上的动态内核调试器（LiveKD）使这些实验运行起来容易且安全。

未涉及到的内容

Windows 2000 是一个庞大而复杂的操作系统。本书并没有涉及到 Windows 2000 内部的每一个方面，只专注介绍基础系统组件。例如本书没有涉及 COM +，它是 Windows 分布式面向对象程序设计设施的基础。

本书只介绍系统内部工作机制，不是针对用户、程序员或系统管理员的，没有讲述如何使用、编程和配置 Windows 2000。

警示和告诫

本书讲述的是 Windows 2000 的内部结构和操作，但很多信息在出版过程中已有所改变（虽然外部界面例如 Win32 API 没有改变）。例如，我们提到了 Windows 2000 系统内部例程、数据结构、内核变量和内部用来作出资源大小和性能相关决定的算法和数值。这些定义的要点在发布过程中可能改变。

并不是说本书中描述的要点都会在出版过程中改变，但我们不能保证它们没有改变。任何使用了没存档的接口的软件都有可能在未来发布的 Windows 2000 上无法工作。更糟的是，使用了没存档的接口的基于内核模式的软件（例如设备驱动器）在 Windows 2000 新的发布版本中可能会导致死机。

配套光盘的使用

这套光盘包括 Sysinternals Web 站点（www.sysinternals.com）的全部内容和其他一些有用的工具。那个网站是由 Mark Russinovich（本书作者之一）和 Bryce Cogswell 负责的。这套光盘也包括本书的电子版以及调试工具和符号（使用调试工具和符号的方法见光盘上的 Readme.txt 文件）。

要查看光盘上的内容，只需把光盘插到光驱里。如果系统有自动运行功能，则画面显示选择项以供查看。如果没有自动运行功能，就从光驱的目录中运行光盘。

系统内部

为了便于用户使用，网站 www.sysinternals.com 的内容已经包括在光盘中，你可以在该网站上找到本书中的实验工具，也可以从光盘上运行这些工具或通过从自动显示画面选择运行安装程序，并按照安装说明把这些工具安装在硬盘上。

可以通过从自动显示画面选择 Browse CD Sysinternals（浏览光盘的 Sysinternals）选项，或者从 Sysinternals - WebSite 目录打开 Ntinternals.htm 文件来查看整个网站内容。可以通过从自动显示画面选择 Run Setup（安装），并根据提示将整个网站的内容拷贝到硬盘中。

要得到最新的 Sysinternals 网站的内容和工具，请访问 www.sysinternals.com 网站，也可以从光盘的自动显示画面选择 Browse Online Sysinternals（浏览在线的 Sysinternals）选项进入该网站。

工具

光盘提供了许多工具，位于 \ Tools 文件夹。这些工具中有性能监视扩展 DLL（KVarPerf），允许你从性能工具中监视 Windows 2000 内核变量。另外还有 LiveKd 工具，它允许标准微软内核调试工具不需要特殊调试选项的情况下运行，微软内核调试工具有 Kd.exe、Windbg.exe、I386Kd.exe 等等。

从自动显示画面中选择 Run Setup（运行安装）选项（或在 \ Setup 文件夹中运行 Setup.exe 文件），并根据提示来安装这些工具，也可以直接从光盘中运行这些工具，然而 LiveKd 必须在光盘上的 Debuggers 目录而不是 Tools 目录下运行。关于光盘的 LiveKd 和系统内核调试的更多信

息请参见光盘根目录下的 Readme.txt 文件)。

系统需求

下列是运行光盘所需要的系统配置:

■ 任何支持的 Microsoft Windows 2000 Professional、Server、Advanced Server 或者 Datacenter Server 配置。

■ 装载实验用的工具 (Tools 和 Sysint 文件夹),需要 5MB 硬盘空间。如果安装 www.sysinternals.com 网站,则需要 30MB 空间。装载 Debuggers 目录下的内容需要 20MB 空间,装载全部 Symbols 内容需要 20MB 空间。

■ 本书中有些实验要用到 Windows 2000 的支持工具、调试工具和源组件 (Professional 版或 Server 版)。第 1 章介绍了这些工具和它们的位置。

E - book

光盘包含本书的电子版,可以在屏幕上阅读本书,查找内容。安装和使用电子版本参见 Ebook 目录下的 Readme.txt 文件。

技术支持

我们做了很大努力以确保书和配套光盘内容的正确性。如果遇到问题,请参考下面的资料。

作者联系方式

本书不是完美的,有可能存在一些不当之处。我们删去了一些本应保留的内容。如果发现错误或提出建议,请给我们发 E - mail 到 insidew2k@sysinternals.com。本书的更新和修改内容会在 www.sysinternals.com/insidew2k 网页上刊出。

微软出版社联系方式

微软公司在下列网站也提供书中错误的正确解答:

http: //mspress.microsoft.com/support/

如果你对本书或配套光盘有什么意见和建议,可与微软出版社联系:

通信地址:

Microsoft Press

Attn: Inside Microsoft Windows 2000 Editor

One Microsoft Way

Redmond, WA 98052 - 6399

E - mail 地址:

mspinput@microsoft.com

上述地址中没有产品支持的信息,访问 www.microsoft.com/windows 2000 网址就可了解微软

公司的产品，也可以在工作日太平洋时间上午 6 点到下午 6 点打电话（425）635 - 7011 与公司联系，或者访问微软的在线支持网站：suppport.microsoft.com/support。

参加本书翻译的有詹剑锋、张文耀、黄艳、魏名、廖彬、巩慧、黄干云、韩华、程小鹏、周武、陈文海、詹向前、刘宏友、魏晓东、袁峰、马健忠、戴思全、叶迪盛、戴亚希、叶海男、陈炳阳、蔡良彬、刘荣、吕可为、王立彬和李小燕。

目　录

第1章　概念和工具

本章介绍 Microsoft Windows 2000 主要的概念和术语，这些概念和术语将在全章使用，如 Microsoft Win32 API、进程、线程、虚拟内存、核心模式与用户模式、对象、句柄、安全以及注册表。我们还将介绍一些可以用来研究 Windows 2000 内部结构的工具，如 Performance 工具、内核调试程序（kernel debugger）、配套 CD 上的特殊工具和各种附加工具包如 Windows 2000 Support Tools、Windows 2000 调试工具、Windows 2000 资源工具包和 Platform Software Development Kit (SDK)。另外，我们还将介绍如何将 Windows 2000 Device Driver Kit（DDK）作为查看 Windows 2000 内部更进一步信息的资源。

一定要理解本章的所有内容，本书的其余章节是基于已经理解了本章内容。

1.1　基础概念和术语

本书中，我们将提到一些读者可能不熟悉的结构和概念。本部分将定义后面使用的术语。在进入后面章节之前你应该熟悉它们。

1.1.1　Win32 API

Win32 应用程序编程接口（application programming interface—API）是 Microsoft Windows 操作系统系列中的主要编程接口，Microsoft Windows 操作系统系列包括 Windows 2000、Windows 95、Windows 98、Windows Millennium Edition 和 Windows CE。尽管在本书中没有描述 Win32 API，但解释了主要 Win32 API 化数的内部行为和实现。有关 Win32 API 编程的全面指导，参见 Jeffrey Richter 的《*Programming Applications for Microsoft Windows*》（第 4 版，微软出版社，1999）。

不同的操作系统实现不同的 Win32 子集。总的来说，Windows 2000 是所有 Win32 实现的超集。哪些服务在哪些相应平台上实现的规定包括在 Win32 API 的参考文档中。该文档可以从 *msdn . microsoft . com* 免费获得，也可以从 MSDN Library CD - ROM 中获得。该文档中的信息在文件 \ Program Files \ Microsoft Platform SDK \ Lib \ Win32api.csv（一个以逗号分隔的文本文件）中也有详细描述，它作为 Platform SDK 的一部分被安装。Platform SDK 来自 MSDN Professional 或从 *msdn . microsoft . com* 免费下载（参见 1.2.6 节）。

注意：MSDN 表示 Microsoft Developer Network，即供微软的开发商使用的支持程序。它有三个 CD - ROM 预订程序：MSDN 库、MSDN 专业版、MSDN 普及版。MSDN 库的内容也可以在 MSDN Web 免费获得。更为详细的信息，参阅 *msdn . microsoft . com*。

在本书中，Win32 API 指基本的函数集，其内容覆盖了进程、线程、存储管理、安全、I/O、窗口和图形等领域。Win32 API 是 Platform SDK 的一部分。Platform SDK 中其他主要内容，如事务处理、数据库、通信、多媒体和网络服务，不在本书中介绍。

尽管 Windows 2000 被设计为支持多个编程接口，但 Win32 是主要的或首选的操作系统接口。这是因为：在三个环境子系统（Win32、POSIX 和 OS/2）中，它对主要的 Windows 2000 系统服务提供了最大程度的访问。第 2 章将介绍，Windows 2000 的应用程序并不直接调用本机的 Windows 2000 系统服务，而必须利用环境子系统提供的 API。

Win32 API 的历史

有趣地是，Win32 并不是 Microsoft Windows NT 的最初编程接口。因为 Windows NT 是作为第二版 OS/2 的替代品开发的，基本的编程接口是 32 位 OS/2 Presentation Manager API。项目开发的一年后，Microsoft Windows 3.0 开始出击市场并获得成功。结果是，微软改变方向，将 Windows NT 作为 Windows 系列产品的替代品，而不是 OS/2 的替代品。这时，需要确定 Win32 API 的开发方向。在此之前，Windows API 仅开发为 16 位接口。

尽管 Win32 API 引进了许多 Windows 3.1 没有的新函数，微软仍决定使新的 API 在函数名字、语义和数据类型的使用上尽可能与 16 位 Windows API 兼容，以减少将已有的 16 位 Windows 应用程序移植到 Windows NT 的负担。因此有些人第一次查看 Win32 API 时，会对一些函数名与接口的不一致感到迷惑，应该记住不一致的一个原因是保证 Win32 API 与旧的 16 位 Windows API 兼容。

1.1.2 服务、函数和例程

Windows 2000 用户和编程文档中的一些术语在不同的上下文中具有不同的意义。例如：服务（service）可以指操作系统中一个可调用的例程、设备驱动程序或服务器进程。下面列出了本书中某些术语的意义。

■ Win32 API 函数。Win32 API 中文档化的、可调用的子例程。如 *CreatProcess*、*CreatFile* 和 *GetMessage*。

■ 系统服务（或执行程序系统服务）。Windows 2000 操作系统中可以从用户模式调用的本机函数（本机函数的定义，请参见 3.1.3 节“系统服务调度”）。例如，*NtCreatProcess* 是 Win32 *CreatProcess* 函数调用的内部系统服务，它创建新的进程。

■ 内核支持函数（或例程）。Windows 2000 操作系统内核模式（本章后面定义）中的子例程。例如，*ExAllocatePool* 是设备驱动程序调用的例程，它从 Windows 2000 系统堆中分配内存。

■ Win32 服务由 Windows 2000 服务控制管理器启动的进程（尽管注册表将 Windows 2000 设备驱动程序定义为“服务”，但在本书中我们并不这样称呼它们）。例如，Task Scheduler 服务是支持 *at* 命令的用户模式进程（它与 UNIX 命令 *at* 或 *cron* 相似）。

■ DLL（动态链接库）。一系列可调用的子例程连接到一起作为二进制文件，可以由使用子例程的应用程序动态装载。例如 Msvcrt.dll（C 运行时库）与 Kernel32.dll（Win32 API 子系统库之一）。Windows 2000 用户模式组件和应用程序广泛使用 DLL。DLL 相对静态库的优点在于应用程序可以共享 DLL，Windows 2000 保证在引用它的应用程序中，只有一个 DLL 代码的内存副本。

1.1.3 进程、线程和作业

尽管程序与进程表面看起来是相似的，但它们在本质上是不同的。程序是一系列静态指令，而进程是由执行程序实例的线程使用的一系列资源的包容器（container）。从最高抽象层上来说，Windows 2000 进程的组成如下：

- 专用的虚拟地址空间，它是进程使用的一系列虚拟存储地址。
- 可执行程序，它定义初始代码和数据，并映射到进程的虚拟地址空间。
- 各种系统资源的开放句柄列表，如信号量、通信端口和文件，它们对进程的所有线程都是可访问的。
- 标识用户并称为“访问令牌”的安全环境、安全组和同进程相关的特权。
- 称为“进程 ID”（内部称为“客户 ID”）的惟一标识符。
- 至少执行一个线程。

线程是进程中 Windows 2000 调度执行的实体。没有线程，进程的程序就不能运行。线程包括下面的必要组件：

- CPU 寄存器的内容，它表示处理器的状态。
- 两个栈，一个是以内核模式执行时被线程使用，另一个则是以用户模式执行时被线程使用。
- 被称为“本机线程存储区”（TLS）的专用存储区，供子系统、运行时库和 DLL 使用。
- 称为“线程 ID”的唯一标识符（内部也称为“客户 ID”，进程 ID 和线程 ID 产生于不同的名字空间，因此它们不会重叠）。
- 线程有时有自己的被多线程服务器应用程序使用的安全环境，多线程服务器应用程序模拟（impersonate）它们服务的客户的安全环境。

易失寄存器，栈和专用存储区被称为线程的环境。因为这些信息对 Windows 2000 运行的每个机器体系结构来说是不同的，这种结构取决于特定体系结构。实际上，由 Win32 GetThreadContext 函数返回的 CONTEXT 结构是唯一与机器相关的 Win32 API 的公共数据结构。

尽管线程有它们自己的执行环境，但进程中的每个线程都共享进程的虚拟地址空间（加上属于进程的其余资源），这意味着进程中的所有线程都可以写入或读取彼此的内存。除非其他的进程把它的专用地址空间的可用部分作为共享内存区域（在 Win32 API 中称为“文件映射对象”），或者一个进程打开另一个进程并使用 ReadProcessMemory 和 WriteProcessMemory 函数，否则线程不能引用另一个进程的地址空间。

除了一个专用地址空间和一个或更多的线程，每个进程还有一个安全标识符和一个对象的开放句柄列表，这些对象可以是文件，共享内存区域或其中的一个同步对象如互斥，事件或信号量，如图 1-1 所示。

每个进程有一个存储在称为“访问令牌”的对象中的安全环境。进程访问令牌包括安全标识符和进程凭证。默认情况下，线程没有自己的访问令牌，但是它们可以得到一个，这样允许单个线程模拟另一个进程的安全环境（包括运行在远程 Windows 2000 系统上的进程），而不影响进程中的其他线程。（关于进程和线程安全的更多细节参见第 8 章。）