Broadview
www.broadview.com.cn

竹林蹊径

——深入浅出Windows驱动开发

驱网核心技术丛书

竹林蹊径

深入浅出 Windows 驱动开发

张佩 马勇 董鉴源 编著

電子工業出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
http://www.phei.com.cn

驱网核心技术丛书

竹林蹊径

——深入浅出Windows驱动开发

深入浅出 Windows 驱动开发

张佩 马勇 董鉴源 编著

電子工業出版社
Publishing House of Electronics Industry
北京•BEIJING

内 容 简 介

本书是作者根据多年的工作学习经验，总结的第一手驱动开发资料。本书更多的是经验之谈，一些实践中的小发现小意外，颇为书中内容添彩。

本书的特色之一，是对WDF框架做了较多的切入。本书第一个主要内容是（第3～7章）围绕WDF而展开讨论，侧重点各有不同。第3章以框架为讨论的中心；第4、5两章以WDF框架开发USB和1394驱动；第6章讲述内核C++编程，也以WDF框架为蓝本；第7章讲述WDF驱动的测试和调试。

第二个主要内容是关于音视频驱动开发（第10～11章）。音视频驱动包括AVStream架构，本书做了较详细的阐述。第10章讲述使用AVStream小端口架构，第11章讲述ASIO音频驱动开发。

第三个主要内容是关于设备驱动安装（第12～14章）。第12章讲系统安装模块，从总体角度阐述系统和设备驱动如何配合完好地进行工作；第13章讲述INF安装文件的细节，包括各个域的作用，以及诸多安装指令的使用。第14章讲如何编写驱动安装软件。

剩余的一些章节，分别是关于驱动入门（第1、2章）、Windbg调试命令（第8章）、内核同步（第9章）等内容。

本书适合一般入门级内核程序员，对WDF有兴趣，准备开发USB或1394设备驱动者，本书尤其有用。本书对于入行较久，经验丰富的程序员，也具有一定的参考价值。

图书在版编目（CIP）数据

竹林蹊径：深入浅出 Windows 驱动开发 / 张佩，马勇，董鉴源编著. —北京：电子工业出版社，2011.3
（驱网核心技术丛书）
ISBN 978-7-121-12555-3

Ⅰ. ①竹… Ⅱ. ①张… ②马… ③董… Ⅲ. ①窗口软件，Windows－驱动程序－程序设计 Ⅳ. ①TP316.7

中国版本图书馆 CIP 数据核字（2010）第 247388 号

责任编辑：李　冰
文字编辑：葛　娜
印　　刷：北京东光印刷厂
装　　订：三河市皇庄路通装订厂
出版发行：电子工业出版社
　　北京市海淀区万寿路 173 信箱　邮编 100036
开　　本：787×1092　1/16　印张：33.25　字数：838 千字
印　　次：2011 年 3 月第 1 次印刷
印　　数：4000 册　定价：69.00 元

凡所购买电子工业出版社图书有缺损问题，请向购买书店调换。若书店售缺，请与本社发行部联系，联系及邮购电话：（010）88254888。

质量投诉请发邮件至 zlts@phei.com.cn，盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线：（010）88258888。

推荐序

我一直认为，编写程序是一件很奇妙的事情，它可以带来创造和控制的欲望。每当我阅读或者编写一段代码时，脑子里自然地就会想象这段代码怎样完成预定的逻辑。当面对一个不熟悉的开发环境，或者一个新的基础平台时，首先要清楚这个环境或者平台是如何工作的，以及提供了哪些功能。代码本身可能非常复杂，甚至奥妙无穷，但通常情况下，真正优美的高质量代码往往是简单的、易于理解的。对于代码编写者或者维护者来说，真正见功夫的地方不在于代码本身，而在于对下层开发平台的理解和驾驭能力，可能这就是俗称的“内功”。

这个观点既适用于应用软件程序员，也适用于系统软件程序员。对于应用软件程序员，低层的应用开发平台是支撑应用开发的基础，譬如，基于 Windows SDK 来开发 Windows 应用程序。那么，程序员有必要理解 Windows SDK 中的基本要素，诸如消息分发机制、各种图形功能等。在这种情况下，阅读一些典型的例子程序代码往往能起到快速引领入门的效果。同样地，C/C++程序员如果局限于 C/C++语言本身，很难编写出高质量的实用程序。他们不仅要掌握 C/C++运行库中函数和类型的用法，甚至还要理解这些函数和类型的实现机理。即使源代码层面上的库，例如 STL（C++的标准模板库），也需要理解其代码实现才能灵活自如地用好这些库（比如 STL 中的各种容器数据结构、迭代器或算法）。

那么，对于系统软件程序员，“内功”是什么呢？系统软件是指操作系统本身或者依附于操作系统上为应用软件提供服务的软件。系统软件可能有机会跟硬件直接打交道，这赋予了程序员更强的控制能力，他们有机会介入操作系统的行为逻辑，甚至改变操作系统的行为特性。但随之而来的是对系统软件代码的更高要求。现代操作系统为应用软件提供了很强的容错能力，应用程序的失败通常不会波及到操作系统自身的稳定性，但操作系统对系统软件的容错能力却比较有限，毕竟系统软件运行起来之后可能被融入到操作系统的执行逻辑中成为操作系统的一部分。因此，理解和掌握操作系统的运行机制成为系统程序员编写出正确、高效的系统软件的基本前提。所谓“内功”，便着落在此。

在 Windows 平台上开发软件，编写 Windows 内核驱动程序是最为考验程序员“内功”的。内核驱动程序的代码量通常不大，但驱动程序框架中的任何一个函数，甚至这些函数中任何一行代码背后都可能蕴含着复杂的逻辑，或者隐式的要求和假设。即使驱动程序编写者在纯粹自行定义的函数中，也必须谨慎地关注一些与环境有关的因素，譬如代码是否可被中断、是否可重入，或者所引用的内存是否被交换到外存。另一方面，应用软件开发中的很多概念，比如地址空间、内存管理、异常处理和多线程并发等，在驱动程序开发中可能需要有不同的理解方法。此外，常用的 C 运行库函数基本上不再适合于驱动程序了，驱动程序编写者必须面对一个全新的底层环境和支持平台。因此，要编写可正确运行的驱动程序，程序员不仅要清楚地理解驱动程序所针对的目标设备或功能（可能包括硬件设备的各种特性），还要掌握 Windows 内核是如何与驱动程序打交道的，以及内核中诸多管理和运行机制，尤其是内存管理、线程调度和并发控制。

当 Windows 内核驱动程序被加载到内核中并且启动以后，它们变成了 Windows 内核的一部分，驱动程序中的接口函数在恰当的时刻被内核调用，这是 Windows 驱动程序的基本工作方式。Microsoft 定义了 WDM（Windows 驱动程序模型）来规定驱动程序的结构，以及 Windows 内核如何与 WDM 驱动程序打交道。WDM 不仅包括 I/O 管理器定义的驱动程序框架，还定义了在驱动程序中如何支持 PnP（Plug and Play，即插即用）、电源管理和 WMI（Windows Management Instrumentation，Windows 管理规范）。因此，若要编写一个完全支持 WDM 的驱动程序，也需要理解 WDM 中所涉及的各个内核组件。

Windows 内核驱动程序与内核的紧密关联性使得驱动程序的调试极为不方便，从某种意义上讲，驱动程序的调试等同于 Windows 内核的调试。而且，对于某些特定的逻辑错误，内核调试器甚至是无能为力的。正因为这个原因，内核驱动程序的代码尽可能精简，从软件设计角度而言，应最大程度地把功能代码放到应用程序中，在驱动程序中只留下最必要的功能逻辑。这样的设计也可以使 Windows 内核被不正确驱动程序代码牵连而导致稳定性问题的几率相对减小。

为了便于 Windows 驱动程序的开发，Microsoft 定义了一个驱动程序框架，称为 WDF（Windows Driver Foundation），其中针对内核驱动程序的部分称为 KMDF（Kernel-Mode Driver Framework）。KMDF 实际上是一个库，它封装了 WDM 中一些基本的代码逻辑，从而使程序员可以更加方便地编写出 WDM 驱动程序。KMDF 可以部分地简化 Windows 内核驱动程序的开发任务，但是本质上它并没有降低内核驱动程序的复杂性，甚至需要程序员付出额外的学习努力。

总而言之，作为一名系统程序员，你需要洞悉目标操作系统中与你的软件打交道的各个部件，也要非常清楚地知道你所依赖的开发工具是如何帮助你做到这一点的。系统程序员往往面临着比应用程序员更长的学习曲线，但是，系统程序员从编写程序中获得的乐趣也是在应用层上难以体会得到的。我相信，当你发现自己编写的软件模块已经与操作系统内核融为一体时，那一刻你的感觉一定是手心里攥着一个操作系统——操作系统尽在你的掌控中了。

这本书《竹林蹊径——深入浅出 Windows 驱动开发》是三位作者张佩、马勇和董鉴源的最新力作，他们将自己在实践中积累起来的经验整理成册，以期望后学者能少走弯路，缩短 Windows 驱动程序开发的学习之路。这本书重点介绍了 KMDF、USB/1394 和音频驱动程序的开发，以及设备驱动程序的发行和安装。建议有一定 Windows 驱动程序开发基础的读者看一看这本书，尤其是，如果你正打算使用 KMDF，或者正在从事与 USB/1394 或音频驱动程序相关的编程工作，那么，这本书便是一份宝贵的实践指导了。

潘爱民

2010 年 12 月 5 日于北京西二旗

推荐序

我认识一个共享软件的作者，近十年来都在开发他的虚拟光驱的软件，不时给这个软件添加一点新的东西。我对此觉得很奇怪，对他说，我觉得虚拟光驱是一个很简单的东西。用一个映象文件容纳光盘上的数据，并开发一套驱动接口让系统以为这是一个光驱。下载网上开源的代码，应该不超过 5000 行。为什么他要为此耗费这么多年的精力呢？

他举了个例子说：国外知名的虚拟光驱 Daemon，它的强项在于兼容性。几乎任何软件都能正常使用它的虚拟光驱，并当做真正的光驱来访问。而普通的开源的虚拟光驱，就有很多不支持了。

他曾经发现一种游戏，要求用户插入光盘。用他自己编写的虚拟光驱来模拟，总是不行。同样的映象文件，换了 Daemon 就一切 OK。这让他大为诧异。碰到这样的情况，他根本就不可能到网上去搜索“为何我的虚拟光驱不支持某某游戏”这样的傻问题，也不可能在某处找来一段代码拷贝一番就解决。他必须找到问题的实质，才能找到对策。

花去漫长的时间，最终分析的结果是，原来因为该游戏希望每个用户都购买正版光盘，它就用了一种特殊的策略来分析用户所用的是不是真实的光驱。大家都知道硬盘的读取速度一般都比较快，而且事实上也更加稳定。光驱读盘的速度比硬盘相对慢一些，而且读取数据的速度有一定的不稳定性。比如说，数据读取的速率可能会以某种数学模型所定义的曲线为轨迹发生波动。而这个软件就根据这种不稳定性的匹配程度来进行检查。如果虚拟光驱提供的数据是不匹配这种特征的，则它很简单地禁止游戏继续运行。

而 Daemon 则在内部插入了这样的模拟函数，有意对数据的读出进行各种延时的处理，使之看上去非常像真实的光驱读出的数据。

总而言之，他开发的是一个逼近世界顶尖品质的好东西。当然代价是汗水与时间。

我能想象到在没有任何公开的代码，或者是前人的经验指引，自己去钻研发现并解决这些问题的困难。能在网上找到解决方案的问题必定不会是软件技术里的关键问题。相反是这样一个一个的无头悬案，才构成了程序员们所谓的“核心技术”。

我和一些人的见解不同。我并不认为越底层的技术就越“核心”。总有人认为系统比软件底层，所以程序更“核心”。而芯片比程序更底层，则芯片又更“核心”。其实硅片比芯片更底层，沙子又比硅片更底层，那是不是沙子才是最核心的技术呢？

我认为，在任何一个领域里，能够进行持之以恒的钻研，当大部分人选择放弃，而你依旧锲而不舍地学习、研究、解决一个又一个实际问题，你就能掌握核心技术。简言之，善于把握自己能够掌握的知识，并不断深化拓展知识领域，这才是真正的学习之道，也是成就个人和团队核心技术的途径。

我见到一些工作过多年的人，很有特点。有一种号称对技术没兴趣，更喜欢做管理，但其实并没有那么多做管理的机会，或者真的有机会，而做得也并不理想；有一种全凭在新手面前吹牛皮支撑老资格，实际编程依旧一塌糊涂。我从来不对别人妄加评论，但从技术学习上讲，他们都算是没到家。

《竹林蹊径——深入浅出 Windows 驱动开发》终于出版了。据我了解，张佩因写作这本书，在家伏案了半年。他是凭着极大的热情和信念去写作的，否则即便薪资上的损失都很值得惋惜。就本书而言，先不去考量书中内容的深浅，技术的精粗，仅就作者的诚心和写作精神，便值得称赞。

牛不是一天吃大的，小牛雏要不断地吃进养料，才能变成大牛。牛人要有牛技术，牛的技术，就是核心技术。我希望这本书的每一位读者，不管你现在或是将来，做的是应用开发还是内核编程，不管用的是 C++还是 Java 或.NET，在工作、学习过程中都具有锲而不舍、精益求精的精神，哪怕最菜鸟的新手，在若干年的积累和沉淀后，都能够逐渐形成自己的核心技术。只有掌握了自己的核心技术，才能进入程序员的自由天地。

此文送给《竹林蹊径——深入浅出 Windows 驱动开发》的读者，是为序。

谭文

2010 年 12 月 10 日

前　言

国内内核开发方面的书籍特别少，一个原因是很多技术牛人，没有时间或机会把自己掌握的知识编辑成书。真的很遗憾。希望我砸出这块砖头后，后面会在书店里看到无数的翡翠之作。

相比较国外的程序员，国内程序员在学习内核驱动开发的时候，学习曲线特别长，主要原因是没有趁手可用的资料。有一些初学者联系我，倒出的苦水大多是：资料太少，技术太难，不知从何下手。我给出的建议多是希望他们努力学习 WDK 中的现成文档，并推荐一些经典的英文电子档。但大多数初级学习者，并不满足我这个答案——他们希望有中文资料。这时候，我会向他们推荐《深入解析 Windows 操作系统》或者《软件调试》，有时候，我还会谨慎地向他们推荐《驱网核心技术丛书》。

很高兴有机会，写成这样一本书。本书的另外两位作者是马勇和董鉴源，他们分别写了第 1 章和第 2 章。写《竹林蹊径——深入浅出 Windows 驱动开发》花了我整整八个月的时间，有半年左右，我把所有的时间都花在上面，不上班，不娱乐。这本书是我的劳苦之作。

我利用写作的机会，纵深渗透学习了不少知识。与其说它是在展示个人才华，不如说是做了一次自我进修和测试。我做不到文采飞扬，能保证的是负责任的态度。此书在写作过程中，增删若干遍，完稿之后，请多位前辈老师审稿。我现在唯一期望的是书中的内容，确实能够对读者起到帮助、参考的作用。

本书特色

《竹林蹊径——深入浅出 Windows 驱动开发》这本书的一个最大特点是插图和示例多，对涉及到的大部分知识，能做到一定深度的挖掘。谭文跟我说，无法把自己的技术经验完全写出来。我当然同意他这句话的正确性；但写作的过程中，我仍然尽最大的可能，把技术和经验文字化、图形化，尽量做到由浅入深，脉络分明——这是我个人的最高目标。

平时给别人讲某个知识点的时候，哪怕是最简单的，我喜欢讲得 360 度面面俱到。

说了一层，爱说下面还有一层。讲了烧水之釜，还要介绍釜底之薪。有人不喜欢这样，但我喜欢。如果以后还写书，我仍会保持这一点。

本书主要内容

本书主要包含这几个方面的内容：WDF 框架、驱动测试、音视频编程、驱动安装。这其中，最费精神的是 WDF 框架相关章节。

WDF 是目前和将来驱动开发的大势所趋。如果初学者因为资料的关系，而紧握着 WDM 架构的双手的话，他一定要留神，不要冷落了旁边正如日中天的 WDF。搞技术，特别是在 Windows 平台上，不建议大伙具有怀旧情绪，跟着形势走是必然的。

书中有四章内容介绍 WDF。笔者如农夫翻地一般，把 WDF 奇异表面下的具体实现做了一定的揭露。笔者饶有兴趣地为大家分析 WDF 的对象模型，而从 Wdf01000 符号文件中揭露的内部结构体定义，能令很多人吃惊不小。

用 WDF 框架编写驱动，要比 WDM 方便、简单一半以上。本书中介绍了使用 WDF 框架进行 USB 和 1394 编程的内容。由于 USB 的运用之广，使 USB 驱动成为 Windows 内核驱动中的显学。驱动开发网专门辟有“USB 驱动”版块，并几乎是最聚人气的地方。为配合 USB 一章的写作与学习，笔者专门请朋友精心设计了一款 USB 驱动学习开发板。读者在本书中的多处地方，都能看到它的玲珑身影。

驱动测试方面包括两章内容，一章以 WDF 驱动测试为中心进行介绍，另一章介绍了 Windbg 调试命令。曾经的王者 SoftIce 湮没不闻后，Windbg 成了唯一的内核调试利器，不可不掌握。

音视频驱动向来都比较小众，做相关开发的公司和个人都很少，资料也就更加少。本书有两章内容介绍音视频开发，一章介绍 AVStream 小端口架构，一章介绍酷酷的 ASIO 音频驱动，并以虚拟 ASIO 声卡的创新技术，为有兴趣的读者带去福音。

本书最后三章，介绍驱动安装有关的知识。一章介绍驱动安装的原理及系统模块，一章介绍 INF 安装文件的技术细节，一章以示例介绍如何编写驱动安装软件。看过这些内容后，试着为你的驱动写一个安装软件，会很酷。

本书读者对象

- 本书适合一般程序员
- 对 WDF 感兴趣，准备开发 VSB 或 1394 设备驱动者

下面要说一些和技术无关的东西。

在本书写作过程中，有许多书外的记忆。比如，夏天我工作的时候，我女儿常常站

在床头，猛地一下用手扑打我的笔记本，电脑屏幕就倾了下去，一阵惊叫。

一次两章隔夜刚新鲜写好的内容，保存在移动硬盘中，第二日在另一台电脑上开机，却怎么都找不到了。翻遍整个系统，用了 N 种数据、磁盘恢复工具，都无济于事，踪影全无。那种无助得想哭的感觉，一直记在心间（这个问题我后来把它再现了，可以认定是 Windows 7 操作系统的 BitLocker 功能在休眠唤醒处理上的一个 Bug）。

致谢

感谢我的好朋友，谭文。是他推荐我主笔这本书的写作。这套系列中的《天书夜读——从汇编语言到 Windows 内核编程》和《寒江独钓——Windows 内核安全编程》出版后，大家都很忙。谭文很信任我，推荐我写《竹林蹊径——深入浅出 Windows 驱动开发》。谭文是我以前的同事，湖南人，故在网上号楚狂人，赫赫有名，散文随笔一级好，技术文章有散文风。他为本书写了一篇小序，特此感谢。谢谢李冰编辑和文字编辑葛娜女士，她们的信任和支持，使我有可能完成这本书。

感谢本书两另外两位作者，他们贡献了第 1 章和第 2 章。

感谢潘爱民老师，他为本书写了序，令我有蓬荜生辉之感。

感谢张银奎老师，张帆兄，他们也对本书给予了鼓励，并写了推荐语。

我要感谢所有照顾过我的亲人们：谢谢我大阿姨，她现在只在天上看着我们。她以力排众议的气势，关心照顾过我。谢谢红兵表哥，他正好大我一轮，学习成绩冠于全镇。当初我老爱从他那里偷书，那些书正是我童年和少年时代仅有的课外书。谢谢小姨父，父执辈中他是唯一给我严肃、客观教育的人。感谢我舅舅，他给我很多帮助。以前，我总是把去上海说成“去我舅舅那”。

谢谢我所有的亲人们。

感谢双方父母，感谢他们所有的辛勤付出。

最后，感谢我的妻子，近两年以来，她离职在家做专职母亲，好像把十几年的事情放在一两年里做了，岁月催人老，我把她累坏了。

最后是一首五言八句，会意书名曰：《竹林蹊径》

荫荫翠竹百亩林，
结庐恒爱此中景。
寻常偶遇方外客，
殷勤指点通幽径。

张佩

目　录

向内核世界说一声：hello，我来了。如果你是一个初学者，并对这个世界充满好奇心，请从这一章开始，我们一起打招呼～

如何在规范的商业环境中，开发成功而有效的驱动软件？驱网站长马勇（ZnSoft）将向你娓娓道来。你会学到这些内容：建立一个简单而有效的开发、调试环境；64 位环境下的内核编程技巧；如何发布你的驱动软件。

WDF 是目前最新的驱动编程框架。当很多内核程序员还紧抱 WDM 的巨大佛脚时，千万要记住，WDF 已是大势所趋。本章介绍了 WDF 最重要的几个概念，并进行了一定程度的深度挖掘。对于 WDF 框架的三大核心模型：对象模型、事件模型、PNP/Power 模型，本章作了重点讲述。

使用 WDF 框架开发 USB 驱动，方便且简单。本章首先总体上从硬件和软件两个方面介绍 USB 相关知识点，包括设备的电气特性、总线结构、USB 驱动类型以及类驱动。编程方面，从 USB 设备初始化、数据操作以及设备控制等几个方面来讲解，透彻并且翔实。

1394 俗称火线。大伙平时最多接触它的地方大概是内核调试时，借助 1394 卡进行双机互联。本章首先从硬件方面介绍了 1394 的知识，它的总线结构很特别，极具可扩性，能非常方便地在各种类型的 1394 设备之间建立数据链路。内核编程方面，本章重点讲解了数据通信相关知识，分为异步通信和同步通信两种方式，颇为复杂，相对难于掌握，但套路是现成的，变化的东西不多，可以熟能生巧。本章最后介绍了 1394 双机互联的原理，有兴趣的读者可参考之。

内核天生适合于 C 语言编程，但越来越多的内核项目，规模达到 10 数万的规模。在这种情况下，人们不由地会将目光投向优雅的 C++语言。总体上说，C 和 C++是至亲好友，内核中使用 C++本不应有什么大问题，但有几个暗礁潜伏已久，不小心的程序员，你可千万不要触礁。

使用 WDF 框架编写的驱动程序，在测试和调试的时候，有特殊的工具。本章介绍了目前所知的三个，它们分别是：Windbg 扩展调试命令、WDFTester 测试工具、WDFVerifier 测试工具。本章将以示例方式，介绍这些工具的使用。

SoftIce 渐行渐远之后，Windbg 成为内核调试的第一利器。使用 Windbg 的最大难点是命令繁多，参数复杂。本章以总结归纳的形式，介绍了作者在工作中经常用到的几大类调试命令，并以实例形式一一介绍。作者根据个人经验所作的分类，未能全备，但能够保证的是，所有实例翔实而可靠，可以作为可信的参考。