第16卷 第4期
2006年4月

计算机技术与发展
COMPUTER TECHNOLOGY AND DEVELOPMENT

Vol.16 No.4
Apr. 2006

WDM驱动程序中设备事件通知技术研究

司 刚,付少锋,周利华
(西安电子科技大学 多媒体技术研究所,陕西 西安 710071)

摘 要:设备驱动程序是应用程序和硬件设备之间通信必不可少的一层,应用程序必须要通过设备驱动程序访问硬件设备。在很多设备的驱动程序中都要求把设备事件,例如中断、数据到达、设备出错等通知给应用程序。因此在驱动程序设计过程中,选择合适的设备事件通知技术至关重要。文中介绍了WDM模型设备驱动程序把设备事件通知给用户模式应用程序的5种方法,说明了每种方法的具体程序设计流程及应注意的事项,并讨论了每种方法的应用范围和局限。

关键词:WDM;驱动程序;应用程序;事件通知
中图分类号:TP311.1 **文献标识码**:A **文章编号**:1005-3751(2006)04-0144-03

Research of Techniques in Device Event Notification in WDM Driver Program

SI Gang, FU Shao-feng, ZHOU Li-hua
(Multimedia Technology Institute, Xidian University, Xi'an 710071, China)

Abstract: Device driver is a necessary layer when an application communicates with a hardware device. In many device drivers, device events, such as interrupt, data arrival and device error should be sent to the applications. Therefore, choosing a proper device event notifying method is vital when designing a device driver. Five methods of notifying the application of some device event are introduced in this paper. Also presents the program design flow and the details that should be paid attention to when using each method in the device driver. In the end, the application domain and limitations of the five methods are presented in this paper.

Key words: WDM; driverprogram; applicationprogram; event notification

0 引 言

在Windows操作系统中,设备驱动程序有着举足轻重的地位,尤其是在WDM(Windows Driver Model)驱动程序模型中,应用程序必须要通过设备驱动程序访问硬件设备,应用程序和硬件设备的通信都要通过处于中间层的设备驱动程序。这种通信是双向的:一是应用程序向设备的通信,这时应用程序可以直接对驱动程序进行调用,由驱动程序在合适的时机把数据或者命令发送到设备驱动程序,这种通信实现起来比较简单;另外一个方向是设备向应用程序的通信,例如设备把准备好的数据发送到应用程序,这就涉及到应用程序如何知道驱动程序已经从设备读取数据,这就需要驱动程序以某种方式把数据准备好这一事件通知给应用程序。

1 通知技术及实现

WDM模型是一个分层的驱动程序模型,较之以往的VxD模型,它带来了很多新的特性,比如支持PNP(Plug and Play),WMI(Windows Management and Instrumentation)和电源管理等。这些新的特性使得WDM模型的驱动程序在把一个设备事件通知给应用程序时有了更多的选择。但是因为WDM驱动程序的设计目标是尽量不需要应用程序的帮助,所以,驱动程序在把一个设备事件通知给应用程序时,VxD模型中的回调函数机制在WDM模型中已经不能使用了[1]。笔者在编写驱动程序过程中,因为需要,对WDM模型设备驱动程序把设备事件通知给应用程序的方法进行了研究,并发现了几种可行的通知技术,它们是:事件法、命名事件法、挂起IRP法、用WMI触发事件法、使用PNP的通知策略法。下面对这几种方法的原理进行详细的说明。

1.1 事件法

这种方法的思想是应用程序用CreatEvent创建一个事件对象,利用一个IOCTL码把这个事件对象的句柄传递给驱动程序,然后调用WaitForSingleObject等待事件的触发[2]。驱动程序在得到事件对象的句柄之后,由事件对象的句柄获得一个指向事件对象的指针,并且保存这个指针,当设备事件发生时,驱动程序触发这个事件,使得因为调用WaitForSingleObject挂起的用户程序继续执行,然后

收稿日期:2005-08-09
作者简介:司 刚(1981-),男,河北迁安人,硕士研究生,研究方向为Windows内核及驱动程序技术;周利华,教授,研究方向为多媒体技术和信息安全技术。

驱动程序删除对事件对象的引用。

图 1 是一个使用该方法的程序的运行模型(里面函数的参数都是示意性的)。

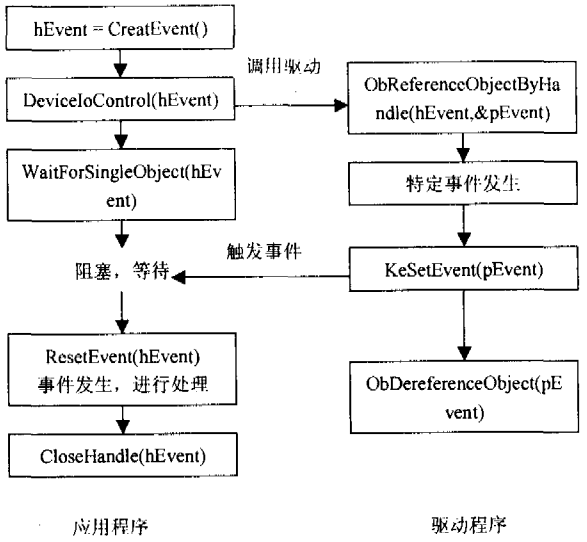

图 1　事件法程序运行模型

需要注意的是,调用 ObReferenceObjectByHandle 函数的例程必须运行于用户应用程序的线程上下文中,因为只有运行在用户应用程序的线程上下文中事件对象的句柄才有效,通常只有文件系统驱动程序、文件系统过滤驱动程序或者单层的驱动程序的 DispatchDeviceControl 例程运行于用户应用程序的线程上下文中。另外调用 KeSetEvent 的例程的 IRQL 必须小于或者等于 DISPATCH-LEVEL,所以不能在中断服务程序里面调用 KeSetEvent,可以在各种延迟过程调用函数里面调用 KeSetEvent。

1.2　命名事件法

这种方法要求核心驱动程序调用 IoCreateNotificationEvent 创建一个命名事件,用户应用程序打开这个命名事件并且等待这个事件。在特定的设备事件发生时,驱动程序调用 KeSetEvent 触发这个事件,使得因为等待这个事件而挂起的用户应用程序继续执行。

图 2 是一个使用该方法的程序的运行模型。

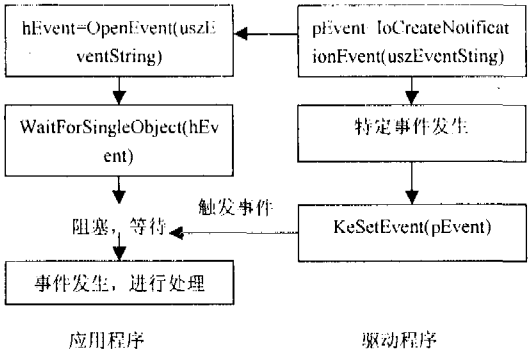

图 2　命名事件法程序运行模型

在 Window 2000 及以后的操作系统中,不能在驱动程序的 DriverEntry 函数里面创建这个命名事件,因为这个时候负责创建 BasedNamedObject 目录的 Win32 子系统还没有加载,而程序中创建的命名事件对象要放在这个目录下。所以,通常在对 CREATE_FILE IRP 的处理函数里面创建命名事件。另外,由于在核心驱动程序中创建的命名事件具有核心态属性,所以,应用程序只能等待这个命名事件,而不能改变它的状态,如果想多次利用这个事件,驱动程序必须负责调用 KeClearEvent()清除这个事件的状态。这种方法在实际应用中使用比较广泛。

1.3　挂起 IRP 法

这种方法要求应用程序向驱动程序发送一个同步或者异步的 IOCTL 请求,然后驱动程序挂起这个 IRP,并且要保存这个 IRP,在某个设备事件发生后,完成这个 IRP,这将使发送同步请求的 DeviceIoControl 函数返回,或者触发一个事件,使得因为发送异步 IOCTL 请求然后等待一个事件的那个线程继续执行。发送异步 IOCTL 请求的时候需要为 DeviceIoControl 函数的最后一个参数提供一个 OVERLAPPED 结构的指针,而且这个函数不必等到 IRP 完成之后才返回,而是立即返回。如果返回值为 ERROR_IO_PENDING,应用程序紧接着调用 WaitForSingleObject 函数等待驱动程序完成这个 IRP[3]。

图 3 是一个使用发送同步 IOCTL 请求的挂起 IRP 法的程序的运行模型(发送异步 IOCTL 请求的挂起 IRP 法程序运行模型这里不再给出)。

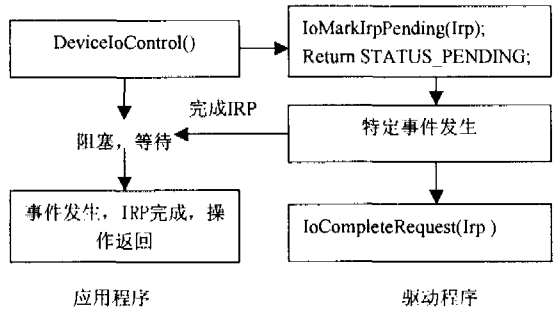

图 3　发送同步 IOCTL 请求的挂起 IRP 法程序运行模型

挂起 IRP 法使得驱动程序的各个例程可以摆脱事件法中的限制,运行于任何线程上下文环境中。同步方法中,在事件触发、函数返回之后,可以立即带回驱动程序传回的数据,而在异步方法中,需要在事件触发之后,再次进行相应的操作取回数据,但是异步方法允许应用程序在注册事件之后,进行其它的操作,然后再等待事件,而且可以指定等待超时值。这使得异步的方法使用更加灵活。

1.4　用 WMI 触发事件法

WMI 是向系统管理员报告管理信息的一种方法,是微软针对 CIM(common information model)模型的一个实现,可以把 WMI 看作是"Windows 上的 CIM"。WMI 的体系结构包括 WMI 提供者、CIM 对象管理器和 WMI 消费者。CIM 对象管理器在这个体系结构的中间负责 WMI 提供者和 WMI 消费者之间的交互。在支持 WMI 的 WDM 驱动程序里面,驱动程序就是一个 WMI 提供者,各种用户应用程序是 WMI 消费者。通过 WMI 类,驱动程序可以通过数据、事件或者方法和应用程序交互[4]。

在驱动程序的 AddDevice 例程里面要调用 IoWMI

RegistrationControl(DeviceObject, WMIREG_ACTION_REGISTER)注册驱动程序成为一个WMI的提供者,然后驱动程序要提供一个DispatchSystemControl例程来处理IRP_MJ_SYSTEM_CONTROL请求。在一定的设备事件发生,而且这个某个注册的事件已经被WMI使能的情况下,驱动程序通过WmiFireEvent触发事件。对于应用程序来说,在连接到WMI之前,必须要先初始化COM,获得一个指向IWbemLocator接口的指针。在连接到WMI之后,就可以用IWbemServices类的ExecNotificationQuery方法进行查询,这个函数立即返回,并且返回的一个指向IEnumWbemClassObject类的指针,可以利用这个类的next方法进行查询,判断事件是否发生。

WMI触发事件法通常是在标准的驱动程序把标准的事件发送给操作系统组件时使用,通常驱动开发者不会使用到这种方法。当然,用户也可以在驱动程序中使用自定义事件。虽然这种通知方法增加了应用程序代码的复杂性,但是它能够在触发事件的同时,把相应的数据发送给应用程序。

1.5 使用PNP通知策略法

WDM驱动程序模型支持PNP,用户程序可以注册标准的PNP事件(例如一个设备的接口被打开),使得特定的PNP事件发生时,能够得到通知,当特定的PNP事件发生时,这些事件会自动触发,无需驱动程序的干预。然而可以利用这种PNP通知策略,在驱动程序里触发自定义的事件。在特定的事件发生时,驱动程序调用IoReportTargetDeviceChangeAsynchronous()触发一个事件,用户应用程序需要调用RegisterDeviceNotification()注册PNP事件,使得当事件发生时,用户的窗口获得一个消息[5]。

这种方法中,特定的事件发生之后,应用程序得到的是一个发送到窗口的消息,这使得应用程序的处理比较简单,但是要求应用程序至少要有一个窗口。应用程序在得到消息的同时,还可以同时得到驱动程序传回来的数据。通常情况下开发USB设备类型的程序的时候使用这种方法比较多。

2 小 结

把设备事件通知给应用程序的方法是设备驱动程序设计中的关键和核心所在。在上述的几种方法中,每一种方法都有其特性和局限,在实际的应用中,要根据自己要编写的驱动程序的类型和需要通知的事件的特性来选择使用。笔者在开发网络中间层驱动程序时,使用了命名事件法,中间层驱动程序截获网络上发送或者接收的数据,通过命名事件通知给应用程序,之后应用程序在共享的缓冲区里面读取发送或者接收的数据并处理;在开发PCI网络安全卡的驱动程序时,使用了挂起IRP法,另外因为程序的逻辑要求在注册一个事件之后,等待这个事件发生之前,还要进行发送数据的操作,所以程序中采用了挂起IRP中的异步法;在开发某USB多媒体键盘驱动程序时,由于要在应用程序中动态监测USB键盘的插拔,故采用了PNP通知策略法。

参考文献:

[1] 张惠娟,周利华,翟鸿鸣.Windows环境下的设备驱动程序设计[M].西安:西安电子科技大学出版社,2002.33-37.

[2] Richter J. Windows核心编程[M].王建华,张焕生,侯丽坤,等译.北京:机械工业出版社,2000.190-227.

[3] Oney W. Programming the Microsoft Windows Driver Model, 2nd Edition Ebook[M]. Washington, USA: Microsoft Press, 2003.

[4] Cant C. Windows WDM设备驱动程序开发指南[M].孙义译.北京:机械工业出版社,2000.204-219.

[5] Microsoft Corporation. Microsoft Windows 2000 Driver Development Kit[EB/OL]. http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dnwinforms/html/wnf-TaskVision.asp,2004.

(上接第143页)

补偿在应用过程中由于各种违规操作所带来的信息泄密等严重后果。在APA系统中采用了Agent技术和数据挖掘技术,从而使其很好地适应了分布式的用户环境,并实时对审计结果做出响应,减少了部门保密信息受到的持续威胁。目前工作重点是引入移动Agent,增加APA系统的智能性和移动性,以更好适应信息安全审计。

目前该系统已经在中国电子科技集团十四所和成都十所、总参六十所等一些军工企业得到了应用,反应良好。

参考文献:

[1] Shoham Y. Agent-Oriented Programming[J]. Artificial Intelligence,1993(60):51-92.

[2] Lee W, Stolfo S, Mok K. Mining Audit Data to Build Intrusion Detection Models[A]. The Fourth International Conference onKnowledge Discovery and Data Mining (KDD'98)[C]. New York, NY:[s.n.],1998.

[3] Sang-Jun Han and Sung-Bae Cho. Detecting intrusion with rule-based integration of multiple models[J]. Computers & Security,2003,22(7):613-623.

[4] Bell D E, LaPadula L J. Secure computer systems: Unified exposition and multics interpretation[R]. Mitre Technical Report ESD-TR-75-306[s.l.]:Mitre Corporation,1976.

[5] Finin T, McKay D, Fritzson R. An Overview of KQML: A Knowledge Query and Manipulation Language[R]. Technical Report, KSL Laboratories,1992.

[6] Finin T, Weber T. Draft Specification of the KQML Agent-Communication Language[EB/OL]. http://www.csee.umbc.edu/kqml/kqmlspec/spec.html,1993-08.