

来随着 VtoolsD 的使用，使用 C 和 C++也开始流行起来。

Windows NT 的设计体现了更现代和模块化的内部体系，它的目标是更好的灵活性和更加的健壮，兼容 Windows3.x 和 Windows9x 并不是它的设计目标，因此，Windows NT 的内部体系中没有提供对以前 Windows 平台的兼容。NT 采用了一种特有的内核模式驱动程序体系，一般也采用 C 语言来编写。当 NT 下的驱动程序需要直接控制机器时，它会向硬件抽象层（HAL）发出请求。硬件抽象层建立在驱动程序和实际的硬件之间，为驱动程序隐藏了硬件的不同，这样就可以编制出跨处理器（比如 Pentium 和 Alpha）、源代码兼容的设备驱动程序。因为 Windows NT 可以工作在单处理器和多处理器环境中，驱动程序必须十分小心的保护关键的数据结构。Windows NT 提供了一种分层的体系结构，每一个 NT 设备驱动程序有一个低层和一个上层接口。低层的驱动程序直接控制硬件。在低层和上层驱动程序之间的是中间层驱动程序。Windows NT 也定义了一种类驱动程序体系，并且支持某些设备类。比如，系统中有 SCSI 类驱动程序支持 SCSI 磁带设备和 SCSI 磁盘。Windows NT 的这种驱动程序体系，在 Windows 98 和 Windows 2000 中得到了继承和扩展，形成了现在的 WDM 体系

认识到跨平台兼容能力的价值后，微软开始尝试统一设备体系，给未来的驱动程序开发提供一个简单的平台。微软的做法，不是重新开发一套新的体系，而是在更合理的 Windows NT 体系的基础上，进行必要的完善，从而形成一个新的设备驱动程序体系，称为 Windows driver model，或 WDM，在更早的技术文档中，微软也曾用过 Win32 Driver Model 这个名字。Windows 98 最先支持 WDM，微软随后推出的操作系统中也都支持 WDM，包括 Windows 2000 系列，Windows Me，Windows XP，但不包括 Win CE。

3.2 WDM 驱动程序

WDM（Windows Driver Model）模型是从 WinNT3.51 和 WinNT4 的内核模式设备驱动程序发展而来的。WDM 主要的变化是增加了对即插即用、电源管理、Windows Management Interface(WMI)、设备接口的支持。WDM 模型的主要目标，是实现能够跨平台使用、更安全、更灵活、编制更简单的 Windows 设备驱动程序。WDM 采用了“基于对象”的技术，建立了一个分层的驱动程序结构。WDM 首先在 Windows98 中实现，在 Windows2000 中得到了进一步的完善，并在后续

开发的 Windows 操作系统中都将存在，比如 Windows Me 和 Windows XP。微软在通过 WDM 模型的引入，希望减轻设备驱动程序的开发难度和周期，逐渐规范设备驱动程序的开发，应该说，WDM 将成为以后设备驱动程序的主流。

3.2.1 WDM 驱动程序模型

WDM 模型主要采用分层的方法，模仿面向对象的技术，按照微软一贯的思路，先进行逻辑上的“分层”，然后将标准的实现和低层细节“封装”起来，形成“基类”，客户程序通过“继承”的方式来扩展“基类”的功能，完成所需要的实现。在微软的技术文献中，称 Windows NT 和 Windows 2000 为“基于对象”（object-based）的系统，和操作系统一样，WDM 驱动程序模型也是“基于对象”的系统。在系统中既使用对象又使用类和继承等机制，而且对象之间仅能通过传递消息实现彼此的通信，这样的方法才称为“面向对象的方法”。如果仅使用对象，则这种方法可以称为“基于对象”的方法。“基于对象”的方法虽然不能得到“面向对象”的所有优点，但依然可以使系统的设计、分析和理解更加的清楚。

在 WDM 模型中，每个硬件设备至少有两个驱动程序：一个功能驱动程序（function driver）和一个总线驱动程序（bus driver）。一个设备还可能有过滤驱动程序（filter driver），用来变更标准设备驱动程序的行为。这些服务于同一个设备的驱动程序组成了一个链表，称为设备栈。

WDM 模型使用了如图 3-1 的层次结构。图中左边是一个设备对象堆栈。设备对象是系统为帮助软件管理硬件而创建的数据结构。一个物理硬件可以有多个这样的数据结构。处于堆栈最底层的设备对象称为物理设备对象(physical device object)，或简称为 PDO。在设备对象堆栈的中间某处有一个对象称为功能设备对象(functional device object)，或简称 FDO。在 FDO 的上面和下面还会有一些过滤器设备对象(filter device object)。位于 FDO 上面的过滤器设备对象称为上层过滤器，位于 FDO 下面(但仍在 PDO 之上)的过滤器设备对象称为下层过滤器。

这样的数据结构。处于堆栈最底层的设备对象称为物理设备对象(physical device object)，或简称为 PDO。在设备对象堆栈的中间某处有一个对象称为功能设备对象(functional device object)，或简称 FDO。在 FDO 的上面和下面还会有一些过滤器设备对象(filter device object)。位于 FDO 上面的过滤器设备对象称为上层过滤器，位于 FDO 下面(但仍在 PDO 之上)的过滤器设备对象称为下层过滤器。