

4.5 USB 驱动程序开发工具

对于开发 WDM 驱动程序来说，我们有以下三个常用组合：

- (1) 直接使用 Windows DDK
- (2) 使用 DriverStudio
- (3) 使用 Windriver

下面我们分别比较三种方式的优缺点。

第一种：开发难度大一些，而且有很多烦琐的工作要作，大部分都是通用的基础性的工作。但如果选用这种方式的话你将对整个体系结构会有很好的理解和把握。

第二种：难度低一些，工具软件已经帮你作了很多基础性的工作。也封装了一些细节，你只要专心去作你需要的操作，但由于封装的问题，可能会带来一些 bug。有可能导致项目的失败。

第三种：几乎没有难度(从开发驱动的角度)。很容易，但只能开发硬件相关的驱动，事实上你写的只是定制和调用它提供的通用驱动而已。效率上有问题。工作频率不是很高。但开发花费的时间很少。是上面的几分之一乃至几十分之一。

一般建议用 windriver 开发驱动程序的原型，用 driverstudio 开发最终发行的驱动程序，如果驱动程序很复杂的话，则直接使用 ddk 开发。

上面的几种情况都需要 VC++ 作为辅助开发环境。前两种情况都需要 ddk。开发时间上，第一种最长，第三种最短，第二种可以认为是前面两种方案的折衷。

4.6 USB 驱动程序设计步骤

4.6.1 USB 驱动程序的编写

在 WDM 驱动程序设计中我们首先需要写一个 DriverEntry 过程，这是每一个设备驱动程序的入口，每次该程序启动时被系统自动调用。在 DriverEntry 中完成驱动程序的初始化工作。在一个 WDM 驱动程序中，初始化是唯一必不可少的。一个驱动程序可以只有初始化段，虽然这样的驱动程序什么也不能做，但它的的确确是一个驱动程序。若要使一个驱动程序能够实现对硬件设备的驱动，还应该有一些其他的回调例程和分发例程来处理各种 IRP，主要有：

DriverUnload

AddDevice