

描述I/O请求的IRP的指针。这个例程可以找到与设备对象联合在一起的驱动程序。在IRP中指定操作类型通常都符合前面讲过的I/O管理器系统调用,例如创建、读取和关闭。

图11-38表示的是一个设备栈在单独一层上的关系。驱动程序必须为每个操作指定一个进入点。IoCallDriver从IRP中获取操作类型,利用在当前级别的设备栈中的设备对象来查找指定的驱动程序对象,并且根据操作类型索引到驱动程序分派表去查找相应驱动程序的进入点。最后会把设备对象和IRP传递给驱动程序并调用它。

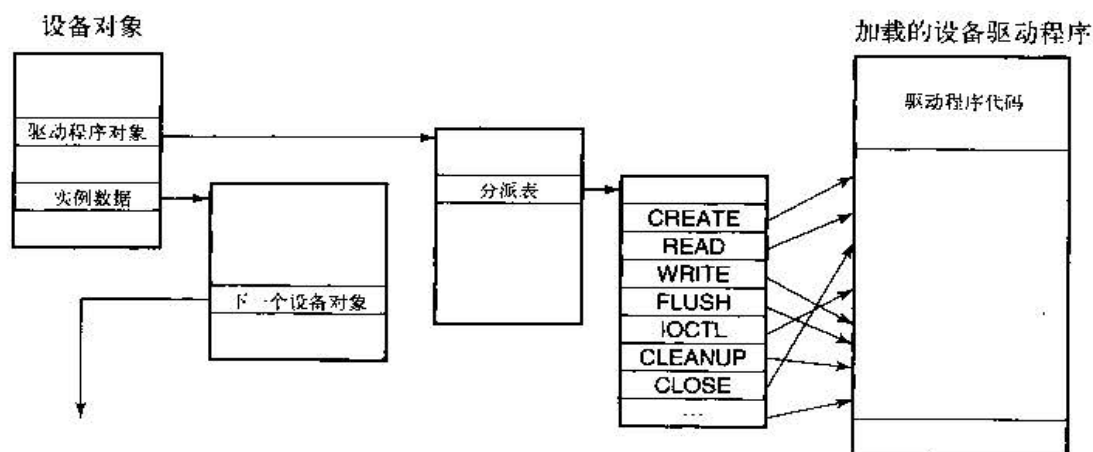


图11-38 设备栈中的单独一层

一旦驱动程序完成处理IRP描述的请求后,它将有三种选择。第一,驱动程序可以再一次调用IoCallDriver,把IRP和设备栈中的下一个设备对象传递给相应的驱动程序。第二,驱动程序也可以声明I/O请求已经完成并返回到调用者。第三,驱动程序还可以在内部对IRP排队并返回到调用者,同时声明I/O请求仍未处理。后一种情况下,如果栈上的所有驱动都认可挂起行为且返回各自的调用者,则会引起一次异步I/O操作。

## 2. I/O请求包

图11-39表示的是IRP中的主要的域。IRP的底部是一个动态大小的数组,包含那些被设备栈管理请求的域,每个驱动程序都可以使用这些域。在完成一次I/O请求的时候,这些设备栈的域也允许驱动程序指定要调用哪个例程。在完成请求的过程中,按倒序访问设备栈的每一级,并且依次调用由每个应用程序指定的完成例程。在每一级,驱动程序可以继续执行以完成请求,也可以因为还有更多的工作要做从而决定让请求处于未处理状态并且暂停I/O的完成。

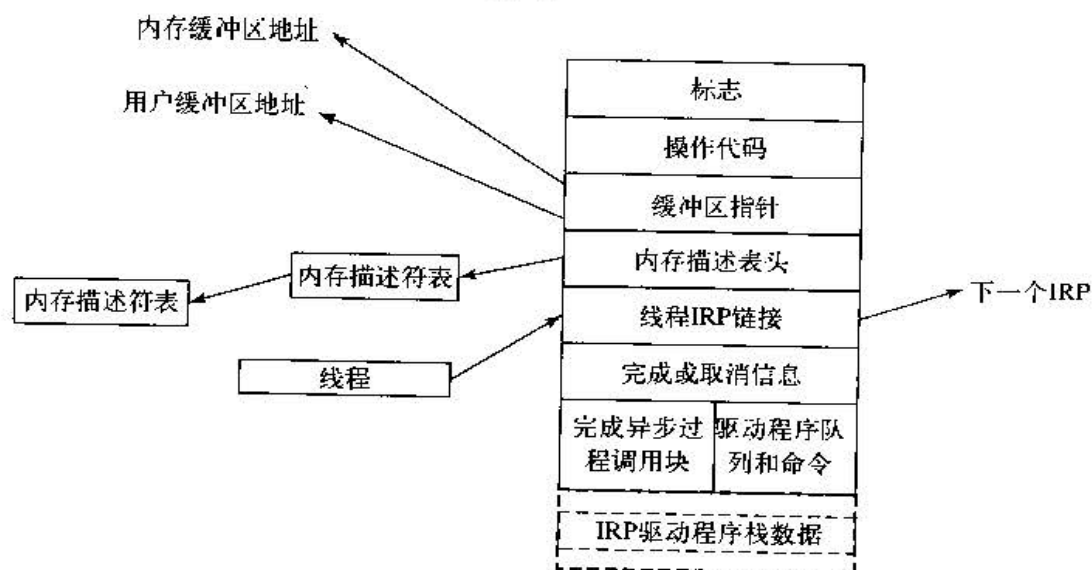


图11-39 I/O请求包的主要域