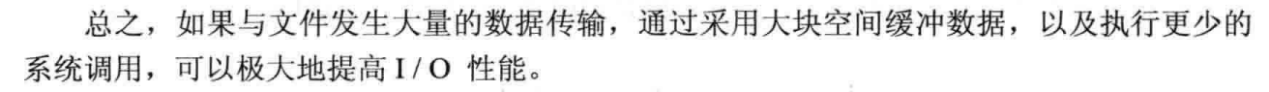
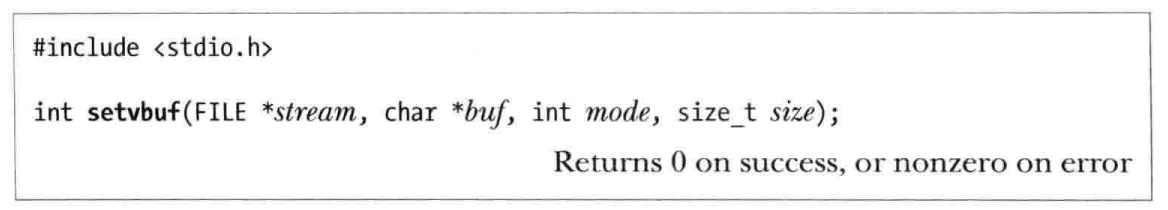
第十三章

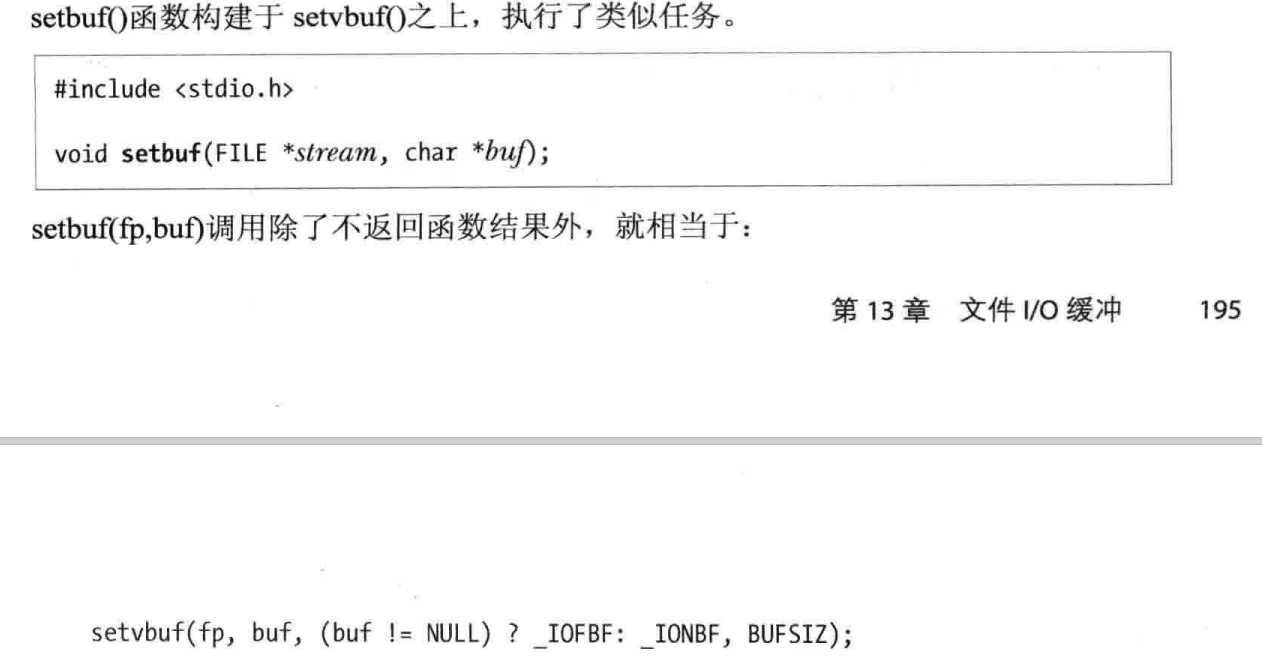
1. 文件IO的内核缓冲：内核缓冲区高速缓存（kernel buffer cache）：read或read系统调用在操作磁盘的时候，不会直接发起磁盘访问，而是仅仅在用户空间缓冲区和内核缓冲区高速缓存之间进行数据交换。现在，操作系统不会单独维护一个缓冲区高速缓存，而是将其置于页面中。
2. 缓冲区大小对IO系统调用性能的影响：

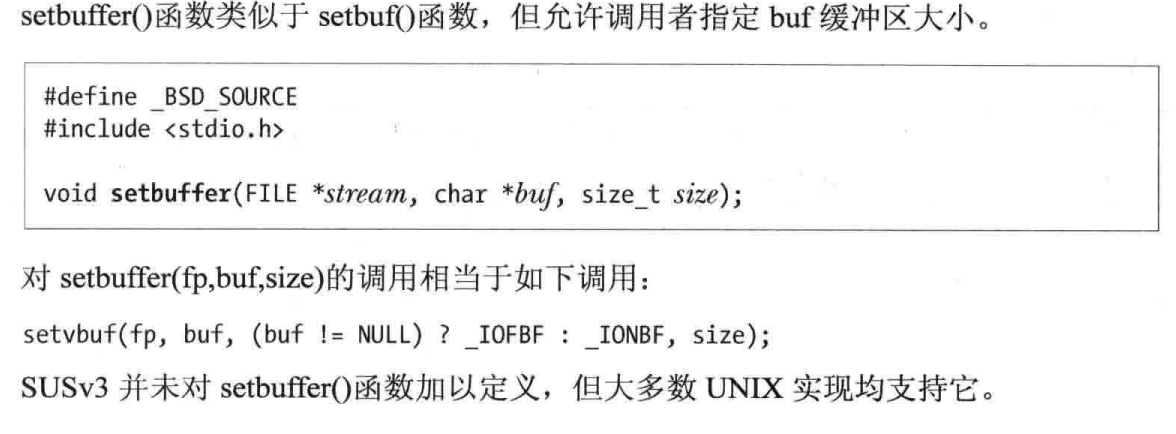


通过实验，write系统调用返回时，尚未将内容复制到磁盘，所以，复制文件大部分时间花费在read读取文件上面。

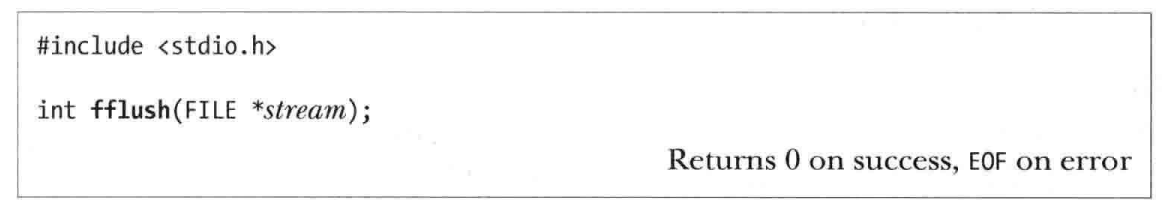
1. Stdio库缓冲：缓冲类型有三种：完全不缓冲，行缓冲，块缓冲。使用一些函数可以调整用户缓冲，从而减少write和read的调用次数。





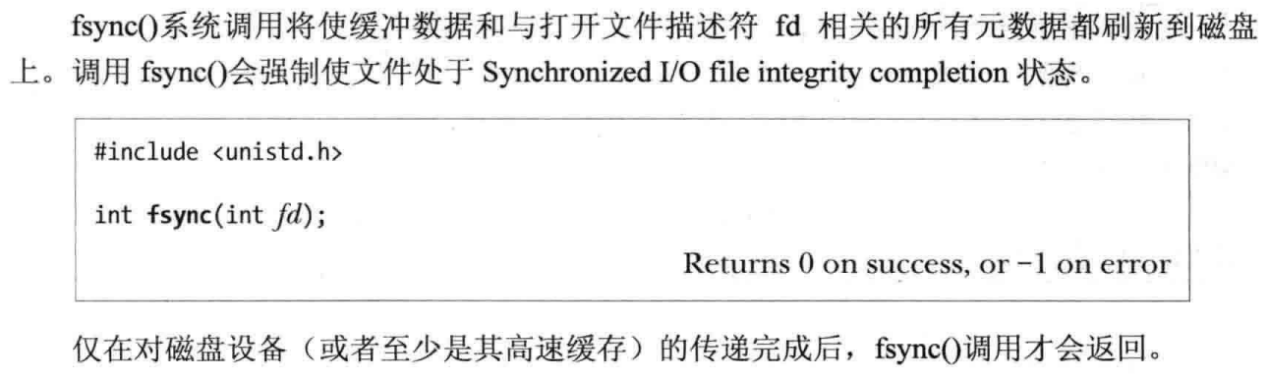


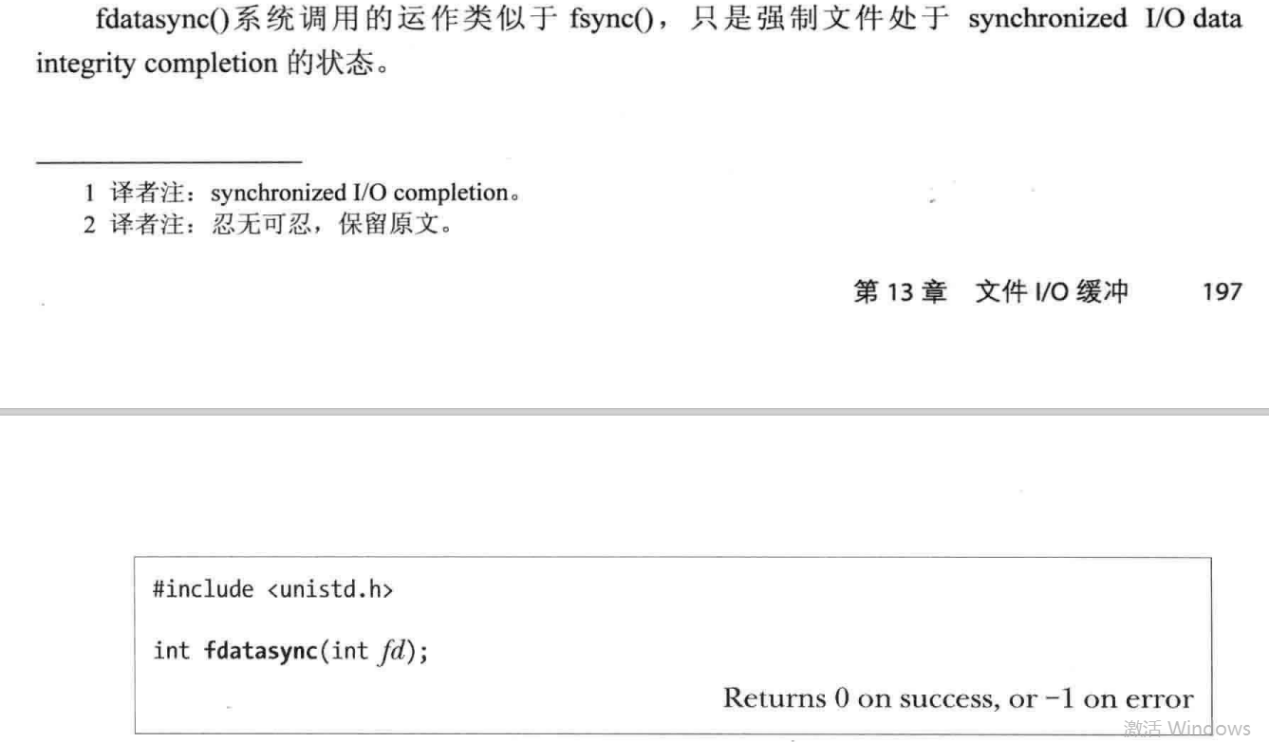
刷新stdio缓冲区：



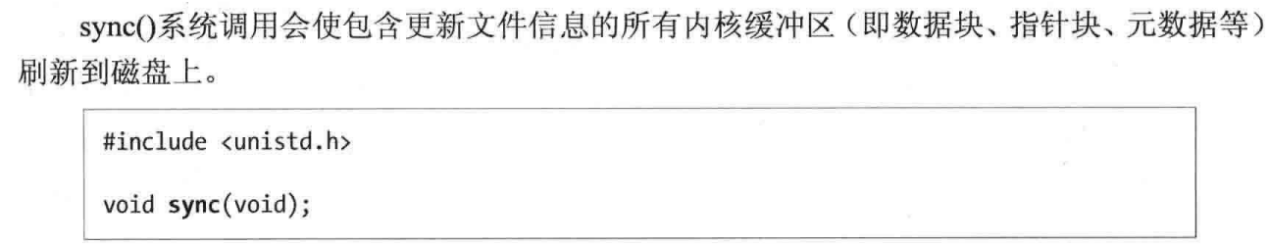
1. 控制文件IO内核缓冲

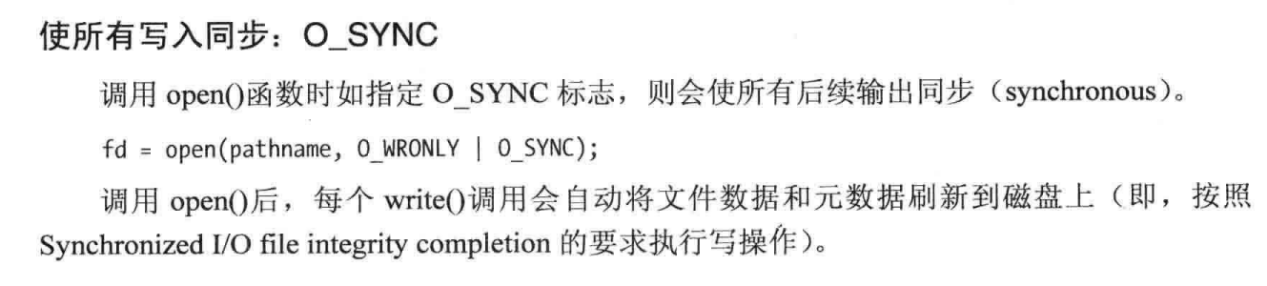
定义：同步IO数据完整性和同步IO文件完整性：区别在于数据写入磁盘后是否立即更新文件元数据。



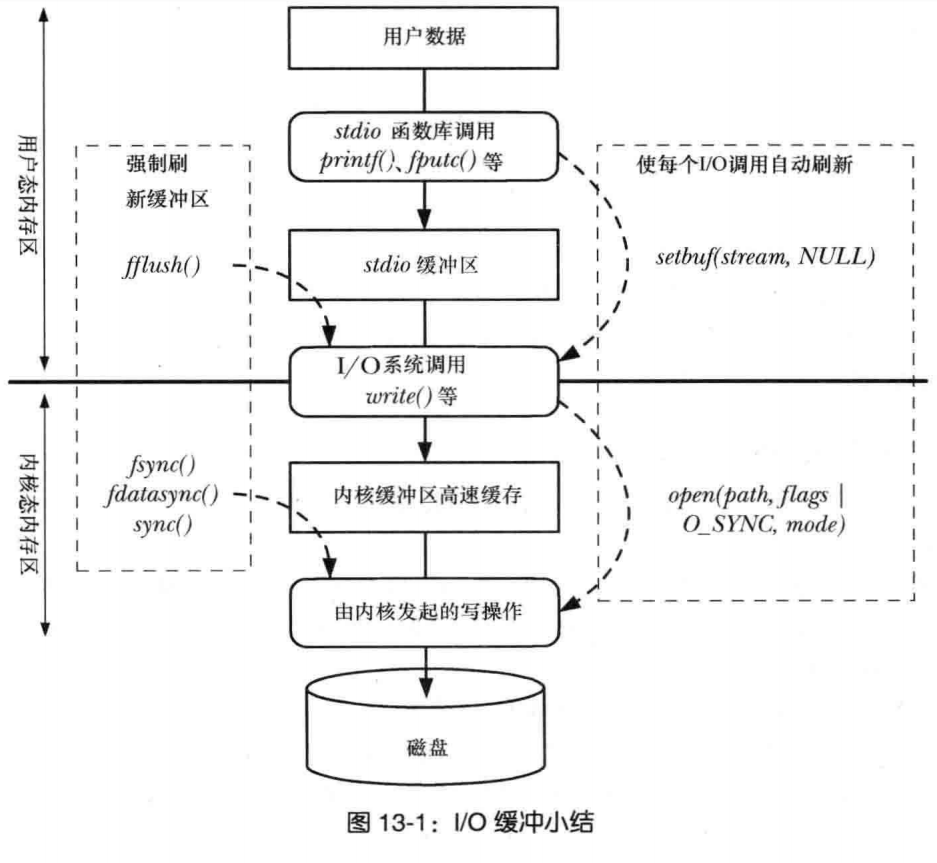


对性能要求极高的，fdatasync调用比fsync要好。

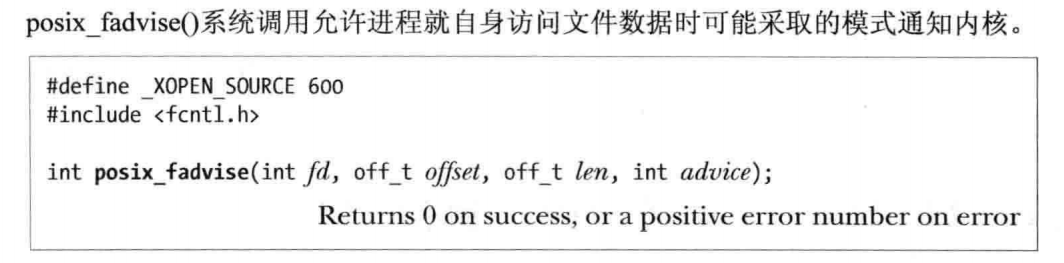




1. IO缓冲小结



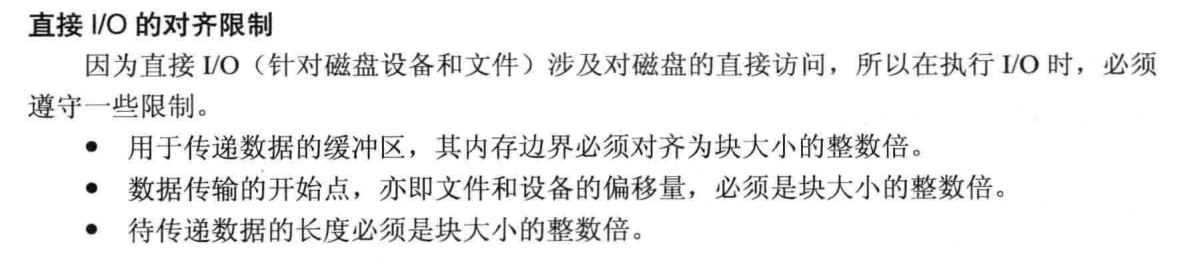
1. 就IO模式向内核提出建议：（个人想法，应该是针对内核缓冲区的预读取相关）：



1. 绕过缓冲区高速缓存：直接IO：

Open函数的O\_DIRECT模式

限制：



1. 混合使用库函数和系统调用：

