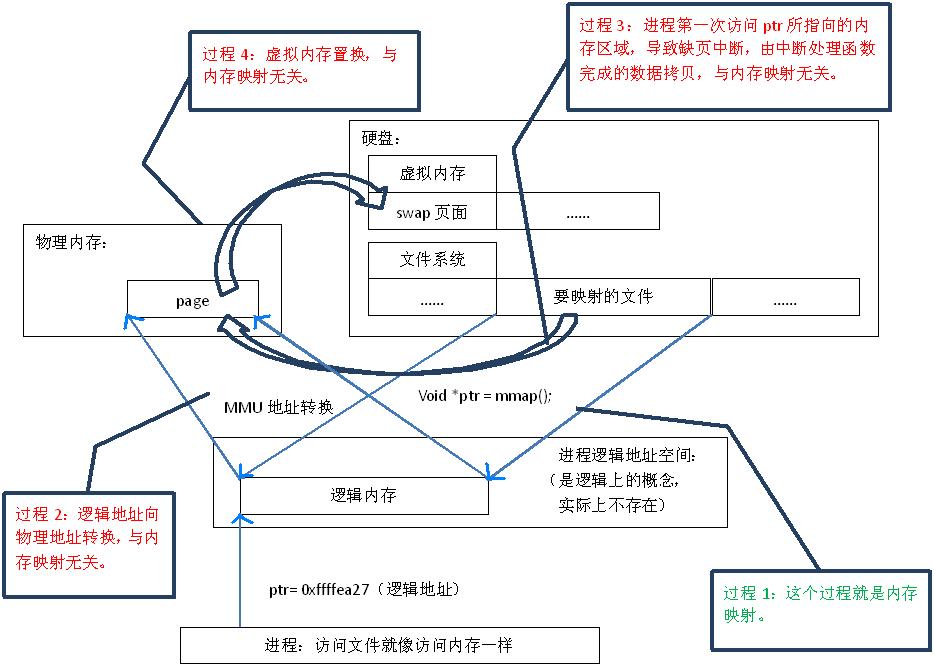


普通文件IO是read和write系统调用， read()是系统调用，其中进行了数据拷贝，它首先将文件内容从硬盘拷贝到内核空间的一个缓冲区，然后再将这些数据拷贝到用户空间，在这个过程中，实际上完成了 **两次数据拷贝。**

****

mmap()也是系统调用，**mmap()中没有进行数据拷贝，真正的数据拷贝是在缺页中断处理时进行的**，由于mmap()将文件从文件系统的地址直接映射到进程（虚拟）用户空间地址，所以中断处理函数根据这个映射关系，直接将文件从硬盘拷贝到用户空间，只进行了 **一次数据拷贝** 。因此，内存映射的效率要比 read/write效率高。

在32位系统上，总的虚拟地址空间4GB，内核占1G，用户空间进程占3G。用户程序访问不了这个1G空间。虚拟地址空间内核部分的内动对于哪个进程来说都是一样的。