

杂乱，未厘清楚

多线程用在 **io** 密集，多进程用在 **cpu** 密集为什么呢？

自我理解一下，如有不对，请脑海删除自己所看及所想，哈哈哈哈

IO密集的程序，说明很多程序在等待使用**IO**，那么用颗粒度较大的线程，可以减少程序的中断次数，所以**IO**密集的程序，用多线程较为合适，且保证程序尽可能的减少被中断

CPU密集的程序，说明很多程序在等待使用**CPU**，用颗粒度较小的进程，切片轮换时间缩短，减少**CPU**排队时间，保证每个程序都可正常运行，防止使用线程模式一个程序长时间占用**CPU**，导致其他程序无法运行引起的性能问题

因为**CPU**的工作频率远远快过和其连接的外部硬件，例如磁盘，所以**CPU**在**IO**的时候经常会需要等待外部硬件完成当前任务，完成之后，才能进行下一个任务，这种情况常常称为**IO**阻塞，即**CPU**直到等待**IO**操作返回前，不能继续运行。**IO**阻塞对于**CPU**强大的运算能力是一个巨大的浪费。

多线程的底层机制是由操作系统实现的，当一个线程遇到**IO**阻塞时，例如读写文件，操作系统可能会暂时挂起该线程，从而让其他线程优先执行，也就是将多出来的时间片切分给其他的线程，直到等待该线程的**IO**操作返回，再重新调度该线程运行。

所以在**IO**密集型任务中使用多线程可以大大的加速程序运行，即时在单核**CPU**上，这种加速主要就是利用了被浪费掉的阻塞时间。