多线程

多线程是指操作系统在单个进程内支持多个并发执行路径的能力。每个进程仅执行单个线程的传统方法称为单线程方法(single-threaded approach)。

在多线程环境中，进程定义为资源分配单元和一个保护单元。与进程相关联的有：

* 容纳进程映像的虚拟地址空间
* 对处理器、其他进程(用于进程间通信)、文件和I/O资源(设备和通道)的受保护访问

一个进程可能有一个或多个线程，每个线程都有：

* 一个线程执行状态(运行、就绪等)
* 未运行时保存的线程上下文；线程可视为在进程内运行的一个独立程序计数器
* 一个执行栈
* 每个线程用于局部变量的一些静态存储空间
* 与进程内其他线程共享的内存和资源的访问

**用户模式调度**(UMS) 是应用程序用于安排自己的线程的一种轻量级机制。应用程序可在不陷入系统调用的情况下，以用户模式切换UMS线程，而且在一个UMS线程在内核中发生阻塞时，程序可以收回对处理器的控制权。每个UMS线程都有自己的线程上下文，它们不共享单个线程的上下文。对于需要极少系统调用的短期工作项目来说，以用户模式在线程之间切换的能力会使得UMS比线程池更有效率。对于需要有效地在多处理器或多核系统上并行多线程这类高性能要求的应用程序来说，UMS模式很有用。要利用UMS的优势，应用程序必须实现调度程序组件，以便管理应用程序的UMS线程并决定何时运行这些线程

**超线程**(Hyper-Threading) 是Intel引入的一项技术，它可以在每个物理核上提供多个逻辑处理器。每个逻辑处理器有自己的CPU状态，但是执行引擎和片上缓存则是共享的。这使得一个逻辑处理器可以在其他逻辑处理器停转(比如缓存未命中，或者分支预测错误)的时候继续执行。调度算法已经被改进过了，原来是讲线程调度到一个空闲的物理处理器你上，现在则改进为“选择一个物理处理器上的空闲逻辑处理器”