

进程和线程的区别

进程：是**并发执行的程序**在**执行过程中分配和管理资源**的基本单位，是一个动态概念，竞争计算机系统资源的基本单位。

线程：是**进程的一个执行单元**，是进程内科调度实体。**比进程更小的独立运行的基本单位**。线程也被称为**轻量级进程**。

一个程序至少一个进程，一个进程至少一个线程。

为什么会有线程？

每个进程都有自己的地址空间，即进程空间，在网络或多用户换机下，一个服务器通常需要接收大量不确定数量用户的并发请求，为每一个请求都创建一个进程显然行不通(系统开销大响应用户请求效率低)，因此操作系统中线程概念被引进。

- 线程的执行过程是线性的，尽管中间会发生中断或者暂停，但是进程所拥有的资源只为改线状执行过程服务，一旦发生线程切换，这些资源需要被保护起来。
- 进程分为单线程进程和多线程进程，单线程进程宏观来看也是线性执行过程，微观上只有单一的执行过程。多线程进程宏观是线性的，微观上多个执行操作。

线程的改变只代表 CPU 的执行过程的变化，而没有发生进程所拥有的资源的变化。

进程线程的区别：

- 地址空间：同一进程的线程共享本进程的地址空间，而进程之间则是独立的地址空间。
- 资源拥有：同一进程内的线程共享本进程的资源如内存、I/O、cpu 等，但是进程之间的资源是独立的。

一个进程崩溃后，在保护模式下不会对其他进程产生影响，但是一个线程崩溃整个进程都死掉。所以多**进程要比多线程健壮**。

进程切换时，消耗的资源大，效率高。所以涉及到频繁的切换时，使用线程要好于进程。同样如果**要求同时进行并且又要共享某些变量的并发操作，只能用线程**不能用进程

- 执行过程：每个独立的进程有一个程序运行的入口、顺序执行序列和程序入口。但是线程不能独立执行，必须依存在应用程序中，由应用程序提供多个线程执行控制。
- 线程是处理器调度的基本单位，但是进程不是。
- 两者均可并发执行。

优缺点：

线程执行开销小，但是不利于资源的管理和保护。线程适合在 SMP 机器（双 CPU 系统）上运行。

进程执行开销大，但是能够很好的进行资源管理和保护。进程可以跨机器前移。

何时使用多进程，何时使用多线程？

对资源的管理和保护要求高，不限制开销和效率时，使用多进程。

要求效率高，频繁切换时，资源的保护管理要求不是很高时，使用多线程。

参考: <http://blog.csdn.net/yaosiming2011/article/details/44280797>

<http://www.cnblogs.com/flashsky/articles/642720.html>