<http://www.ruanyifeng.com/blog/2013/04/processes_and_threads.html>

进程与线程的一个简单解释

Threads and Locks

Threads in java:

Implements java.lang.Runnable interface

Extends Thread class

Public interface Runnable {

Void run();

}

Synchronized

Lock.lock();中间放的代码是同步的

Deadlock,

Two thread are waiting for the object lock that another one holds.

A deadlock need

1. mutual exclusion limited quantity of resource

2. hold and wait

3. No preemption: One process cannot forcibly remove another process’ resource.

4. Circuit wait.

Processes and threads are related to each other but are fundamentally differet.

Process is an instance of a program in execution, which is an independent entity to CPU/MEM system allocated. 互相独立在不同的address space, 如果要通信必须IPC， shared mem， MPI， sockets， pipe， file

Thread exists within a process and shares the process’s resoures( including its heap space).

Thread 共享heap space，each thread still has its own registers and own stack, and it can read and write the heap memory.

Thread 之间是好兄弟，修改process的时候互相可见

**1) Explain the main purpose of an**[**operating system**](http://career.guru99.com/category/operating-system-2/)**?**

Operating systems exist for two main purposes. One is that it is designed to make sure a computer system performs well by managing its computational activities. Another is that it provides an environment for the development and execution of programs.

**2) What is demand paging?**

Demand paging is a system wherein area of memory that are not currently being used are swapped to disk to make room for an application’s need.

**3) What are the advantages of a multiprocessor system?**

With an increased number of processors, there is considerable increase in throughput. It can also save more money because they can share resources. Finally, overall reliability is increased as well.

**4) What is kernel?**

Kernel is the core of every operating system. It connects applications to the actual processing of data. It also manages all communications between software and hardware components to ensure usability and reliability.

**5) What are real-time systems?**

Real-time systems are used when rigid time requirements have been placed on the operation of a processor. It has well defined and fixed time constraints.

**6) What is virtual memory?**

Virtual memory is a memory management technique for letting processes execute outside of memory. This is very useful especially is an executing program cannot fit in the physical memory.

**7) Describe the objective of multiprogramming.**

The main objective of multiprogramming is to have process running at all times. With this design, CPU utilization is said to be maximized.

**8 ) What are time sharing systems?**

In a Time sharing system, the CPU executes multiple jobs by switching among them, also known as multitasking. This process happens so fast that users can actually interact with each program while it is running.

**9) What is SMP?**

SMP is short for Symmetric MultiProcessing, and is the most common type of multiple-processor systems. In this system, each processor runs an identical copy of the operating system, and these copies communicate with one another as needed.

**10) How are**[**server**](http://career.guru99.com/category/server/)**systems classified?**

Server systems can be classified as either computer-server systems or file server systems. In the first case, an interface is made available for clients to send requests to perform an action. In the second case, provisions are available for clients to create, access and update files.

**11) What is asymmetric clustering?**

In asymmetric clustering, a machine is in a state known as hot standby mode where it does nothing but to monitor the active server. That machine takes the active server’s role should the server fails.

**12) What is a thread?**

A thread is a basic unit of CPU utilization. In general, a thread is composed of a thread ID, program counter, register set and the stack.

**13) Give some benefits of multithreaded**[**programming**](http://career.guru99.com/category/programming-2/)**.**

– there is an increased responsiveness to the user  
– resource sharing within the process  
– economy  
– utilization of multiprocessing architecture

**14) Briefly explain FCFS.**

FCFS is short for First-come, first-served, and is one type of scheduling algorithm. In this scheme, the process that requests the CPU first is allocated the CPU first. Implementation is managed by a FIFO queue.

**15) What is RR scheduling algorithm?**

RR (round-robin) scheduling algorithm is primarily aimed for time-sharing systems. A circular queue is setup in such a way that the CPU scheduler goes around that queue, allocating CPU to each process for a time interval of up to around 10 to 100 milliseconds.

**16) What necessary conditions can lead to a deadlock situation in a system?**

Deadlock situations occur when four conditions occur simultaneously in a system: Mutual exclusion; Hold and Wait; No preemption; and Circular wait.

**17) Enumerate the different RAID levels.**

RAID 0 – Non-redundant striping  
RAID 1 – Mirrored Disks  
RAID 2 – Memory-style error-correcting codes  
RAID 3 – Bit-interleaved Parity  
RAID 4 – Block-interleaved Parity  
RAID 5 – Block-interleaved distributed Parity  
RAID 6 – P+Q Redundancy

**18) Describe Banker’s algorithm**

*[](http://career.guru99.com/wp-content/uploads/2012/05/Bankers_Algorithm.gif)*

*Bankers Algorithm*

Banker’s algorithm is one form of deadlock-avoidance in a system. It gets its name from a banking system wherein the bank never allocates available cash in such a way that it can no longer satisfy the needs of all of its customers.

**19) What factors determine whether a detection-algorithm must be utilized in a deadlock avoidance system?**

One is that it depends on how often a deadlock is likely to occur under the implementation of this algorithm. The other has to do with how many processes will be affected by deadlock when this algorithm is applied.

**20) Differentiate logical from physical address space.**

Logical address refers to the address that is generated by the CPU. On the other hand, physical address refers to the address that is seen by the memory unit.

**21) How does dynamic loading aid in better memory space utilization?**

With dynamic loading, a routine is not loaded until it is called. This method is especially useful when large amounts of code are needed in order to handle infrequently occurring cases such as error routines.

**22) What are overlays?**

Overlays are used to enable a process to be larger than the amount of memory allocated to it. The basic idea of this is that only instructions and data that are needed at any given time are kept in memory.

**23) What is the basic function of paging?**

Paging is a memory management scheme that permits the physical-address space of a process to be noncontiguous. It avoids the considerable problem of having to fit varied sized memory chunks onto the backing store.

**24) What is fragmentation?**

Fragmentation is memory wasted. It can be internal if we are dealing with systems that have fixed-sized allocation units, or external if we are dealing with systems that have variable-sized allocation units.

**25) How does swapping result in better memory management?**

[](http://career.guru99.com/jobs/?utm_source=career.guru99&utm_medium=referral&utm_campaign=click)

During regular intervals that are set by the operating system, processes can be copied from main memory to a backing store, and then copied back later. Swapping allows more processes to be run that can fit into memory at one time.

**26) Give an example of a Process State.**

– New State – means a process is being created  
– Running – means instructions are being executed  
– Waiting – means a process is waiting for certain conditions or events to occur  
– Ready – means a process is waiting for an instruction from the main processor  
– Terminate – means a process is done executing

**27) What is a socket?**

A socket provides a connection between two applications. Each endpoint of a communication is a socket.

**28) What is Direct Access Method?**

Direct Access method is based on a disk model of a file, such that it is viewed as a numbered sequence of blocks or records. It allows arbitrary blocks to be read or written. Direct access is advantageous when accessing large amounts of information.

**29) When does thrashing occur?**

Thrashing refers to an instance of high paging activity. This happens when it is spending more time paging instead of executing.

**30) What is the best page size when designing an operating system?**

The best paging size varies from system to system, so there is no single best when it comes to page size. There are different factors to consider in order to come up with a suitable page size, such as page table, paging time, and its effect on the overall efficiency of the operating system.

**31) When designing the file structure for an operating system, what attributes are considered?**

Typically, the different attributes for a file structure are naming, identifier, supported file types, and location for the files, size, and level of protection.

**32) What is root partition?**

Root partition is where the operating system kernel is located. It also contains other potentially important system files that are mounted during boot time.

**33) What are device drivers?**

Device drivers provides a standard means of representing I/O devices that maybe manufactured by different companies. This prevents conflicts whenever such devices are incorporated in a systems unit.

**34) What are the primary functions of VFS?**

VFS, or Virtual File System, separates file system generic operations from their implementation by defining a clean VFS interface. It is also based on a file-representation structure known as vnode, which contains a numerical designator needed to support network file systems.

**35) What are the different types of CPU registers in a typical operating system design?**

– Accumulators  
– Index Registers  
– Stack Pointer  
– General Purpose Registers

**36) What is the purpose of an I/O status information?**

I/O status information provides info about which I/O devices are to be allocated for a particular process. It also shows which files are opened, and other I/O device state.

**37) What is multitasking?**

Multitasking is the process within an operating system that allows the user to run several applications at the same time. However, only one application is active at a time for user interaction, although some applications can run “behind the scene”.

**38) What are some pros and cons of a command line interface?**

A command line interface allows the user to type in commands that can immediately provide results. Many seasoned computer users are well accustomed to using the command line because they find it quicker and simpler. The main problem with a command line interface is that users have to be familiar with the commands, including the switches and parameters that come with it. This is a downside for people who are not fond of memorizing commands.

**39) What is caching?**

Caching is the processing of utilizing a region of fast memory for a limited data and process. A cache memory is usually much efficient because of its high access speed.

**40) What is spooling?**

Spooling is normally associated with printing. When different applications want to send an output to the printer at the same time, spooling takes all of these print jobs into a disk file and queues them accordingly to the printer.

**41) What is an Assembler?**

An assembler acts as a translator for low level language. Assembly codes, written using mnemonic commands are translated by the Assembler into machine language.

**42) What are interrupts?**

Interrupts are part of a hardware mechanism that sends a notification to the CPU when it wants to gain access to a particular resource. An interrupt handler receives this interrupt signal and “tells” the processor to take action based on the interrupt request.

**43) What is GUI?**

GUI is short for Graphical User Interface. It provides users with an interface wherein actions can be performed by interacting with icons and graphical symbols. People find it easier to interact with the computer when in a GUI especially when using the mouse. Instead of having to remember and type commands, users just click on buttons to perform a process.

**44) What is preemptive multitasking?**

Preemptive multitasking allows an operating system to switch between software programs. This in turn allows multiple programs to run without necessarily taking complete control over the processor and resulting in system crashes.

**45) Why is partitioning and formatting a prerequisite to installing an operating system?**

Partitioning and formatting creates a preparatory environment on the drive so that the operating system can be copied and installed properly. This includes allocating space on the drive, designating a drive name, determining and creating the appropriate file system structure.

**46) What is plumbing / piping?**

It is the process of using the output of one program as an input to another. For example, instead of sending the listing of a folder or drive to the main screen, it can be piped and sent to a file, or sent to the printer to produce a hard copy.

**47) What is NOS?**

NOS is short for Network Operating System. It is a specialized software that will allow a computer to communicate with other devices over the network, including file/folder sharing.

**48) Differentiate internal commands from external commands.**

Internal commands are built-in commands that are already part of the operating system. External commands are separate file programs that are stored in a separate folder or directory.

**49) Under DOS, what command will you type when you want to list down the files in a directory, and at the same time pause after every screen output?**  
**a) dir /w  
b) dir /p  
c) dir /s  
d) dir /w /p**

Answer: d) dir /w /p

**50) How would a filenamed EXAMPLEFILE.TXT appear when viewed under the DOS command console operating in Windows 98?**

The filename would appear as EXAMPL~1.TXT . The reason behind this is that filenames under this operating system is limited to 8 characters when working under DOS environment.

**A condition variable** is just a boolean or integer variable used in an if statement. It's is suitable for single threaded logic but not sufficient for communication between threads. Because:  
  
1) If two threads try to change a variable at the same time it will blow up the process....  
  
2) If you are suspending a thread with a conditional statement like this

Code:

while ( i ); // < waiting until i is cleared before thread will continue

It's just about the worst thing you can do for performance. Your CPU will just grind until i is reset.. likely starving other processes and thread of cpu cycles.  
  
**A Semephore** is a conditional variable suited for inter thread communications. A semephore can be used with specific functions to block thread execution without using cpu clock cycles. A semapore is a stop light which tells a thead to go or stop in a cpu friendly manor.  
  
**A mutex** like a semaphore is cpu cycles friendly. Only a mutex doens't act like a stop light. It acts as a gate, or yeild sign. If two threads are headed for collision when they reach the mutex, one will be paused just long enough to divert the desaster rather than blocking on a semephore until the semephore is cleared.

Condition Variables 和锁定机制有所不同（Mutexes 等），它提供了另外一种同步的机制。例如，当某个线程拥有 mutex 时，mutex 被用于让其他的线程等待（直到拥有线程释放此 mutex），相比较来说，一个 condition variable 则被用于让当前线程等待，直到其他线程激发（signal）此 condition variable 后线程才继续执行。可以同时存在多个线程等待某个 condition variable 的情况。Mutex is like a token

条件变量是让此线程等待， mutex让别人等待。

假定我们存在变量 ok，必须在其为 true 时当前线程才继续向下执行，而 ok 会被其他线程修改。如果没有 condition variables，那么我们必须在当前线程中使用轮询的方式（polling）。

Condition variable 提供了一种让线程等待直到某个条件出现的机制。条件详细来说则是指某个变量满足某个条件，例如上面说到的变量 ok 值为 true，这里我们把这种变量叫做条件相关变量（这里之所以不叫做条件变量是为了避免和 condition variable 同步机制这一术语混淆）。条件相关变量会在不同线程中使用，因此同步机制 condition varaible 总是和 mutex 一起使用，此 mutex 用于保护条件相关变量。

**Conditional variable is essentially a wait-queue**, that supports blocking-wait and wakeup operations, i.e. you can put a thread into the wait-queue and set its state to BLOCK, and get a thread out from it and set its state to READY.

Note that to use a conditional variable, two other elements are needed:

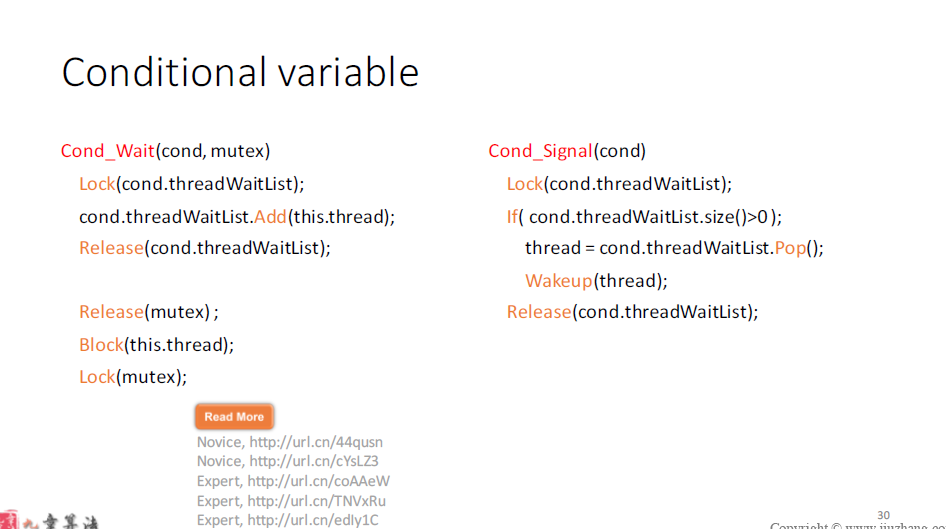
* a condition (typically implemented by checking a flag or a counter)
* a mutex that protects the condition

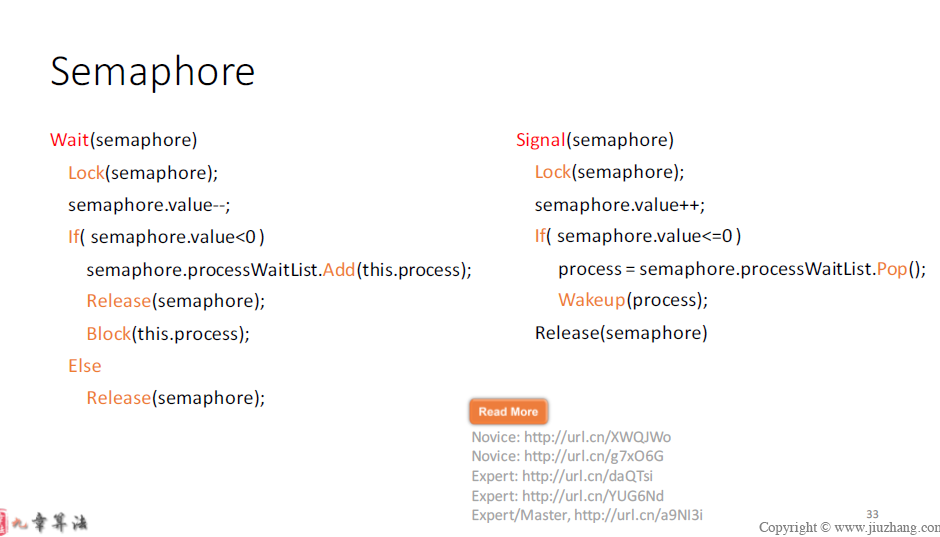
The protocol then becomes,

1. acquire mutex
2. check condition
3. block and release mutex if condition is true, else release mutex

**Semaphore is essentially a counter + a mutex + a wait queue.** And it can be used as it is without external dependencies. You can use it either as a mutex or as a conditional variable.

Therefore, semaphore can be treated as a more sophisticated structure than conditional variable, while the latter is more lightweight and flexible.





**non-preemptive multitasking**, is a style of [computer multitasking](https://en.wikipedia.org/wiki/Computer_multitasking) in which the [operating system](https://en.wikipedia.org/wiki/Operating_system) never initiates a[context switch](https://en.wikipedia.org/wiki/Context_switch) from a running [process](https://en.wikipedia.org/wiki/Process_(computing)) to another process.

In contrast, [preemptive](https://en.wikipedia.org/wiki/Preemption_(computing)) multitasking interrupts applications and gives control to other processes outside of the application's control.

面经问题， cache， process 和 thread，linux可以算thread吗，

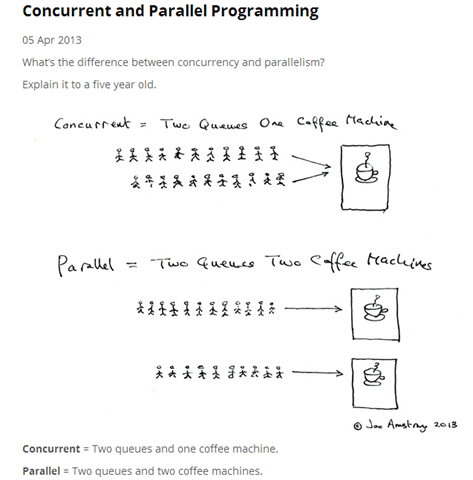
linux执行一个程序的时候具体是怎么搞的，实现细节

### **并发和并行，异步与多线程区别**

发表于2015/4/21 9:01:50  416人阅读

分类： Java学习笔记

1、并发和并行的区别



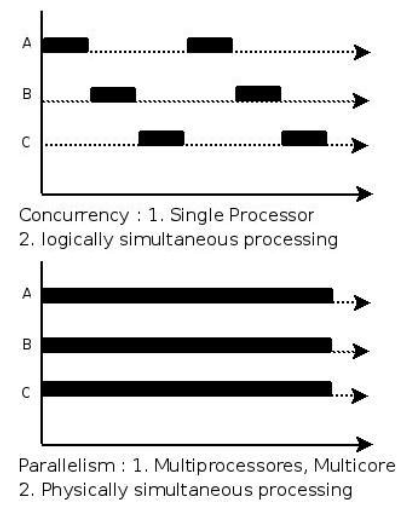
可由上图形象指出两者的区别：

1）定义：

并发：在[操作系统](http://baike.baidu.com/view/880.htm)中，是指一个时间段中有几个程序都处于已启动运行到运行完毕之间，且这几个程序都是在同一个[处理机](http://baike.baidu.com/view/2107226.htm)上运行，但任一个时刻点上只有一个程序在处理机上运行。

并行：在操作系统中，一组程序按独立异步的速度执行，无论从微观还是宏观，程序都是一起执行的。

来个比喻：并发和并行的区别就是一个人同时吃三个馒头和三个人同时吃三个馒头；



在单CPU系统中，系统调度在某一时刻只能让一个线程运行，虽然这种调试机制有多种形式(大多数是时间片轮巡为主)，但无论如何，要通过不断切换需要运行的线程让其运行的方式就叫并发(concurrent)。而在多CPU系统中，可以让两个以上的线程同时运行，这种可以同时让两个以上线程同时运行的方式叫做并行(parallel)。

2）并发通常指提高运行在单处理器上的程序的性能；

并发是有状态的，“具有可论证的确定性，但是实际上具有不可确定性”；

"并发"在微观上不是同时执行的，只是把时间分成若干段，使多个进程快速交替的执行，从宏观外来看，好像是这些进程都在执行。

使用多个线程可以帮助我们在单个处理系统中实现更高的吞吐量，如果一个程序是单线程的，这个处理器在等待一个同步I/O操作完成的时候，他仍然是空闲的。在多线程系统中，当一个线程等待I/O的同时，其他的线程也可以执行。

二、异步与多线程

1）基本概念

    1. 并发：在[操作系统](http://baike.baidu.com/view/880.htm)中，是指一个时间段中有几个程序都处于已启动运行到运行完毕之间，且这几个程序都是在同一个[处理机](http://baike.baidu.com/view/2107226.htm)上运行。其中两种并发关系分别是同步和互斥  
    2. 互斥：进程间相互排斥的使用临界资源的现象，就叫互斥。  
    3. 同步：进程之间的关系不是相互排斥临界资源的关系，而是相互依赖的关系。进一步的说明：就是前一个进程的输出作为后一个进程的输入，当第一个进程没有输出时第二个进程必须等待。具有同步关系的一组并发进程相互发送的信息称为消息或事件。  
    其中并发又有伪并发和真并发，伪并发是指单核处理器的并发，真并发是指多核处理器的并发。  
    4. 并行：在单处理器中多道程序设计系统中，进程被交替执行，表现出一种并发的外部特种；在多处理器系统中，进程不仅可以交替执行，而且可以重叠执行。在多处理器上的程序才可实现并行处理。从而可知，并行是针对多处理器而言的。并行是同时发生的多个并发事件，具有并发的含义，但并发不一定并行，也亦是说并发事件之间不一定要同一时刻发生。

    5. 多线程：多线程是程序设计的逻辑层概念，它是进程中并发运行的一段代码。多线程可以实现线程间的切换执行。

    6. 异步：异步和同步是相对的，同步就是顺序执行，执行完一个再执行下一个，需要等待、协调运行。异步就是彼此独立,在等待某事件的过程中继续做自己的事，不需要等待这一事件完成后再工作。线程就是实现异步的一个方式。异步是让调用方法的主线程不需要同步等待另一线程的完成，从而可以让主线程干其它的事情。

    异步和多线程并不是一个同等关系,异步是最终目的,多线程只是我们实现异步的一种手段。异步是当一个调用请求发送给被调用者,而调用者不用等待其结果的返回而可以做其它的事情。实现异步可以采用多线程技术或则交给另外的进程来处理。

    异步和同步的区别，  在io等待的时候，同步不会切走，浪费了时间。

    多线程的好处，比较容易的实现了 异步切换的思想， 因为异步的程序很难写的。多线程本身程还是以同步完成，但是应该说比效率是比不上异步的。 而且多线很容易写， 相对效率也高。

2）深层次理解

多线程和异步操作的异同

　　多线程和异步操作两者都可以达到避免调用线程阻塞的目的，从而提高软件的可响应性。甚至有些时候我们就认为多线程和异步操作是等同的概念。但是，多线程和异步操作还是有一些区别的。而这些区别造成了使用多线程和异步操作的时机的区别。

异步操作的本质

　　所有的程序最终都会由计算机硬件来执行，所以为了更好的理解异步操作的本质，我们有必要了解一下它的硬件基础。 熟悉电脑硬件的朋友肯定对DMA这个词不陌生，硬盘、光驱的技术规格中都有明确DMA的模式指标，其实网卡、声卡、显卡也是有DMA功能的。DMA就是直接内存访问的意思，也就是说，拥有DMA功能的硬件在和内存进行数据交换的时候可以不消耗CPU资源。只要CPU在发起数据传输时发送一个指令，硬件就开始自己和内存交换数据，在传输完成之后硬件会触发一个中断来通知操作完成。这些无须消耗CPU时间的I/O操作正是异步操作的硬件基础。所以即使在DOS这样的单进程（而且无线程概念）系统中也同样可以发起异步的DMA操作。

线程的本质  
　　线程不是一个计算机硬件的功能，而是操作系统提供的一种逻辑功能，线程本质上是进程中一段并发运行的代码，所以线程需要操作系统投入CPU资源来运行和调度。

异步操作的优缺点

　　因为异步操作无须额外的线程负担，并且使用回调的方式进行处理，在设计良好的情况下，处理函数可以不必使用共享变量（即使无法完全不用，最起码可以减少共享变量的数量），减少了死锁的可能。当然异步操作也并非完美无暇。编写异步操作的复杂程度较高，程序主要使用回调方式进行处理，与普通人的思维方式有些初入，而且难以调试。

多线程的优缺点  
　　多线程的优点很明显，线程中的处理程序依然是顺序执行，符合普通人的思维习惯，所以编程简单。但是多线程的缺点也同样明显，线程的使用（滥用）会给系统带来上下文切换的额外负担。并且线程间的共享变量可能造成死锁的出现。

适用范围

　　在了解了线程与异步操作各自的优缺点之后，我们可以来探讨一下线程和异步的合理用途。我认为：当需要执行I/O操作时，使用异步操作比使用线程+同步I/O操作更合适。I/O操作不仅包括了直接的文件、网络的读写，还包括数据库操作、Web Service、HttpRequest以及.Net Remoting等跨进程的调用。  
　　而线程的适用范围则是那种需要长时间CPU运算的场合，例如耗时较长的图形处理和算法执行。但是往往由于使用线程编程的简单和符合习惯，所以很多朋友往往会使用线程来执行耗时较长的I/O操作。这样在只有少数几个并发操作的时候还无伤大雅，如果需要处理大量的并发操作时就不合适了。