**进程线程通信方式之间的差异**

**每个进程有自己的地址空间。**两个进程中的地址即使值相同，实际指向的位置也不同。进程间通信一般通过操作系统的公共区进行。

**同一进程中的线程因属同一地址空间，可直接通信。**

不仅是系统内部独立运行的实体，而且是独立竞争资源的实体。

线程也被称为轻权进程，同一进程的线程共享全局变量和内存，使得线程之间共享数据很容易也很方便，但会带来某些共享数据的互斥问题。

许对程序为了提高效率也都是用了线程来编写。

父子进程的派生是非常昂贵的，而且父子进程的通讯**需要ipc或者其他方法**来实现，比较麻烦。而线程的创建就花费少得多，并且同一进程内的线程共享全局存储区，所以通讯方便。

线程的缺点也是由它的优点造成的，主要是同步，异步和互斥的问题，值得在使用的时候小心设计。

只有进程间需要通信,同一进程的线程share地址空间,没有通信的必要，但要做好同步/互斥mutex,保护共享的全局变量。线程拥有自己的栈。同步/互斥是原语primitives.  
而进程间通信无论是信号，管道pipe还是共享内存都是由操作系统保证的，是系统调用.

线程间通信：由于多线程共享地址空间和数据空间，所以多个线程间的通信是一个线程的数据可以直接提供给其他线程使用，而不必通过操作系统（也就是内核的调度）。

进程间的通信则不同，它的数据空间的独立性决定了它的通信相对比较复杂，需要通过操作系统。

以前进程间的通信只能是单机版的，现在操作系统都继承了基于套接字（socket）的进程间的通信机制。这样进程间的通信就不局限于单台计算机了，实现了网络通信。

**一、进程间的通信方式**

进程间的通信，它的数据空间的独立性决定了它的通信相对比较复杂，需要通过操作系统。以前进程间的通信只能是单机版的，现在操作系统都继承了**基于套接字（socket）的进程间的通信机制**。这样进程间的通信就不局限于单台计算机了，实现了网络通信。  
 进程的通信机制主要有：**管道、有名管道、信号、信号量、消息队列、共享内存、套接字。**

# 管道( pipe )：管道是一种半双工的通信方式，数据只能单向流动，而且只能在具有亲缘关系的进程间使用。**进程的亲缘关系通常是指父子进程关系。**  
# 有名管道 (namedpipe) ： 有名管道也是半双工的通信方式，但是它允许**无亲缘关系进程间的通信。**# 信号 (sinal ) ： 信号是**一种比较复杂的通信方式，用于通知接收进程某个事件已经发生**。信号是进程间通信机制中唯一的异步通信机制，可以看作是异步通知.  
# 信号量(semophore ) ： **信号量是一个计数器，**可以用来**控制多个进程对共享资源的访问**。它常作为一种锁机制，防止某进程正在访问共享资源时，其他进程也访问该资源。因此，**主要作为进程间以及同一进程内不同线程之间的同步手段。**# 消息队列( messagequeue ) ： **消息队列是由消息的链表，存放在内核中并由消息队列标识符标识。**消息队列克服了信号传递信息少、管道只能承载无格式字节流以及缓冲区大小受限等缺点。  
# 共享内存(shared memory ) ：共享内存就是映射一段能被其他进程所访问的内存，**这段共享内存由一个进程创建，但多个进程都可以访问。共享内存是最快的 IPC 方式**，它是针对其他进程间通信方式运行效率低而专门设计的。它往往与其他通信机制，如信号两，配合使用，来实现进程间的同步和通信。  
# 套接字(socket ) ： 套解口也是一种进程间通信机制，与其他通信机制不同的是，**它实现了**实现了网络通信**。**

**二、线程间的通信方式**

1. 锁机制：包括互斥锁、条件变量、读写锁   
   * 互斥锁提供了以排他方式防止数据结构被并发修改的方法。
   * 读写锁允许多个线程同时读共享数据，而对写操作是互斥的。
   * 条件变量可以以原子的方式阻塞进程，直到某个特定条件为真为止。对条件的测试是在互斥锁的保护下进行的。条件变量始终与互斥锁一起使用。
2. 信号量机制(Semaphore)：包括无名线程信号量和命名线程信号量   
   信号机制(Signal)：类似进程间的信号处理

线程间的通信目的主要是用于线程同步，所以线程没有像进程通信中的用于数据交换的通信机制。