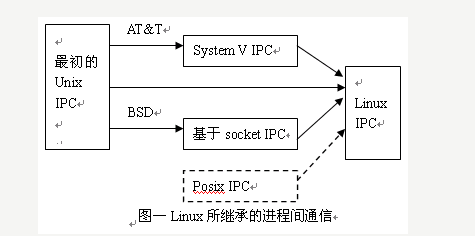
**进程之间的通信总纲**

**作者：chw**

**一.进程间的所有通信方式简介**

**linux下的进程通信手段基本上是从Unix平台上的进程通信手段继承而来的。而对Unix发展做出重大贡献的两大主力AT&T的贝尔实验室及BSD（加州大学伯克利分校的伯克利软件发布中心）在进程间通信方面的侧重点有所不同。前者对Unix早期的进程间通信手段进行了系统的改进和扩充，形成了“system V IPC”，通信进程局限在单个计算机内；后者则跳过了该限制，形成了基于套接口（socket）的进程间通信机制。Linux则把两者继承了下来，如图示：**



**最初的Unix IPC包括：有名管道，无名管道，信号。；System V IPC包括：System System V共享内存区、V消息队列、System V信号灯。由于Unix版本的多样性，电子电气工程协会（IEEE）开发了一个独立的Unix标准，这个新的ANSI Unix标准被称为计算机环境的可移植性操作系统界面（PSOIX）。现有大部分Unix和流行版本都是遵循POSIX标准的，而Linux从一开始就遵循POSIX标准。**

**二.常用的进程间的通信方式**

**<1>管道（Pipe）及有名管道（named pipe）：管道可用于具有亲缘关系进程间的通信，有名管道克服了管道没有名字的限制，因此，除具有管道所具有的功能外，它还允许无亲缘关系进程间的通信；**

**<2>信号（Signal）：信号是比较复杂的通信方式，用于通知接受进程有某种事件发生，除了用于进程间通信外，进程还可以发送信号给进程本身**

**<3>消息队列:** **消息队列是消息的链接表，包括Posix消息队列system V消息队列。有足够权限的进程可以向队列中添加消息，被赋予读权限的进程则可以读走队列中的消息。消息队列克服了信号承载信息量少，管道只能承载无格式字节流以及缓冲区大小受限等缺点。**

**<4>共享内存：使得多个进程可以访问同一块内存空间，是最快的可用IPC形式。是针对其他通信机制运行效率较低而设计的。往往与其它通信机制，如信号灯结合使用，来达到进程间的同步及互斥。**

**<5>信号灯集（semaphore）：主要作为进程间以及同一进程不同线程之间的同步手段。**

**<6>套接口（Socket）：更为一般的进程间通信机制，可用于不同机器之间的进程间通信。常用于网络编程中。**