

## 进程

### 标识符

- 进程标识符：简称pid，是进程的唯一标识。
- 父进程标识符：简称ppid，标识该进程的父进程，即创建进程的进程所对应的pid。
- 用户标识符：简称uid，标识创建该进程的用户。此外euid标识有效用户标识符。
- 组标识符：简称gid，标识创建进程用户的所属组。Euid对应的组标识符即为egid。

### 进程状态

通常进程的状态被划分为五种：初始态、就绪态、运行态、睡眠态和终止态。初始态一般不进行讨论，因为当初始化完成后，进程会立刻转化为就绪态。

#### ①就绪态

处于就绪态（Ready）的进程，所需的其它资源已分配到位，此时只等待cpu，当可以使用cpu时，进程会立刻变为运行态。

#### ②运行态

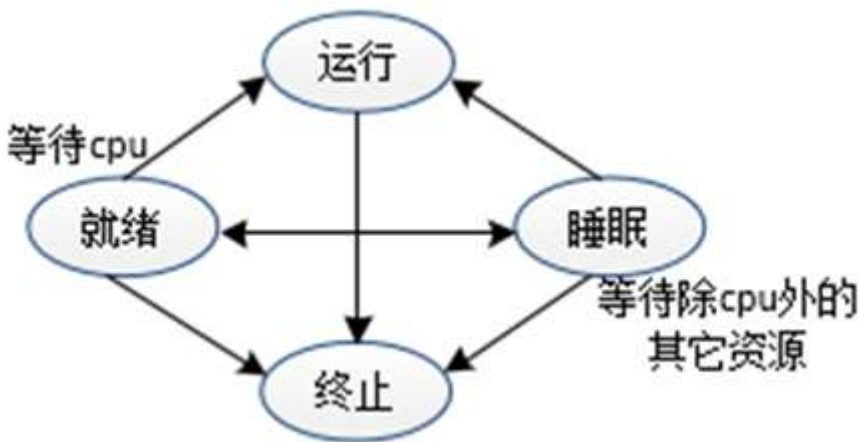
进程处于运行态（Execting）时会占用cpu，处于此状态的进程的数目必定小于等于处理器的数目，即每个cpu上至多能运行一个进程。

#### ③睡眠态

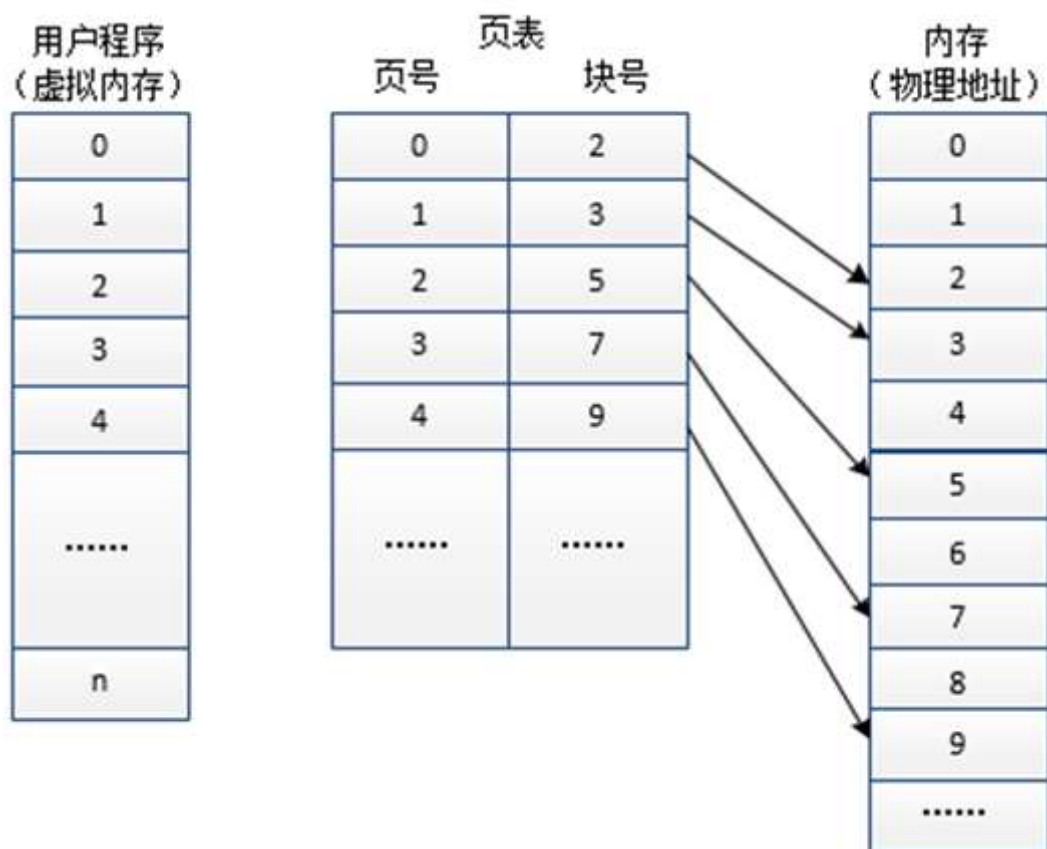
处于睡眠态（Sleeping）的进程会因某种原因而暂时不能占有cpu。睡眠态分为不可中断的睡眠和可中断的睡眠。

#### ④终止态

处于终止态的进程已运行完毕，此时进程不会被调度，也不会再占用CPU。



每个进程的pcb中都有一个指向页表的指针，进程、页表与内存之间的映射关系如图所示。



进程组由用户启动的进程创建，用户启动的进程是进程组中的领导进程（process group leader）。进程组中领导进程的pid亦是识别进程组的进程组id，即pgid。

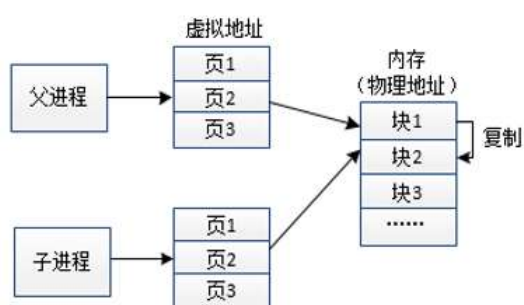
## 进程组与会话

会话（session）是进程组的集合，会话中的每个进程组称为一个工作(job)。

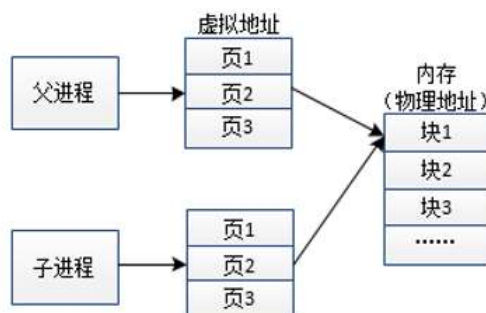
会话由其中的进程创建，创建会话的进程称为会话的领导进程(session leader)，会话领导进程的pid也是标识会话的会话id，即sid。

一个会话中一般有一个进程组工作在前台，使用终端，其余进程组工作在后台（在终端执行命令时，在命令之后添加“&”则将命令启动的进程放在后台执行）。会话的意义在于可在同一个终端执行多个进程组。

## 进程控制



写时复制



读时共享

fork()

exec函数族

exit() \_exit()

exit()和\_exit()都是用来终止进程的，但它们所做的操作有些许差别：当程序执行到\_exit()函数时，系统会无条件地停止剩下操作，终止进程，并清除进程所用内存空间以及进程在内核中的各种数据结构；exit()函数对\_exit()进行了包装，在执行退出前还有若干道工序，最重要的就是它会在调用\_exit()之前先检查文件的打开情况，将缓冲区中的内容写回文件。相对而言，exit()函数相比\_exit()函数更为安全。

①孤儿进程

父进程在进程退出前退出，子进程变为孤儿进程。

②僵尸进程

当进程调用了exit()函数后，该进程并不是马上消失，而是留下一个称为僵尸进程（Zombie）的数据结构。僵尸进程是Linux系统中的另一种特殊进程，它几乎放弃进程退出前占用的所有内存，既没有可执行代码，也不能被调度，只在进程列表中保留一个位置，记载进程的退出状态等信息，供父进程收集。

进程同步

wait()

处于僵尸态的进程不能再次被运行，但是却会占用一定的内存空间，并占据进程编号，当系统中僵尸进程较多时，将会消耗系统的大部分内存，新的进程可能因内存不足或无法获取pid而无法创建，因此应尽量避免僵尸进程的产生。

在父进程中通过wait()和waitpid()函数可以有效防止僵尸进程的产生，对于内存中已经存在的僵尸进程，则可通过杀死其父进程的方法解决。

当僵尸进程的父进程被终止后，僵尸进程将作为孤儿进程被init进程接收，init进程会不断调用wait()函数获取子进程状态，对已处于僵尸态的进程进行处理。

孤儿进程永远不会成为僵尸进程。

进程管理命令

ps

ps是“Process Status”的缩写。在命令行输入“ps”后回车就能查看当前系统中正在运行的进程。ps的命令格式如下：

ps [选项] [参数]

选项	说明
a	显示当前终端机下的所有进程，包括其它用户启动的进程
u	以用户的形式，显示系统中的进程
x	忽视终端机，显示所有进程
e	显示每个进程使用的环境变量
r	只列出当前终端机中正在执行的进程

## top

ps命令执行后，会显示执行命令那一刻系统中进程的相关信息，若想使信息动态地显示，可以使用top命令。top的命令格式如下：

top [选项]

top命令可以实时观察系统的整体运行情况，默认时间间隔为3秒，即每3秒更新一次页面，类似Windows系统中的任务管理器，是一个很实用的系统性能监测工具。

## pstree

pstree命令以树状图的形式显示系统中的进程，可以直接观察到进程之间的派生关系。pstree命令的格式如下：

pstree [选项]

## pgrep

pgrep命令根据进程名从进程队列中查找进程，查找成功后默认显示进程的pid。pgrep命令的格式如下：

pgrep [选项] [参数]

## nice

nice命令用于设置Linux系统中进程的nice值，该命令的格式如下：

nice [选项] [参数]

## bg fg

bg命令的作用是，将进程放入后台运行，使前台可以执行其它任务。其命令格式如下：

bg 参数

fg命令的作用与bg相反，是将后台的进程调往前台。其命令格式如下：

fg 作业号

## jobs

使用jobs命令可以查看Linux系统中的作业列表及作业状态，进程中的作业也有编号，编号从1开始。Linux系统中作业从用户角度进行编号，进程从系统管理员的角度进行编号。jobs命令的语法格式如下：

jobs [选项] [参数]

当选项和参数缺省时，默认显示作业编号、作业状态和启动作业的命令。

## kill

kill命令用来终止正在运行的进程，它的工作原理是发送某个信号到指定进程，以终止该进程。kill的命令格式如下：

kill [选项] [参数]