



进程管理

计算机学院 盛剑会

进程管理

进程

Process

是当程序被触发后，执行者的权限与属性、程序的代码与所需数据等都会被加载到内存中，操作系统给予这个内存内的单元一个**标识符** (Process ID, 简称PID)。

程序

Procedure

通常为binary program，存放在计算机的硬盘、光盘、U盘等存储媒体中，以实体文件的形态存在。



进程管理

1

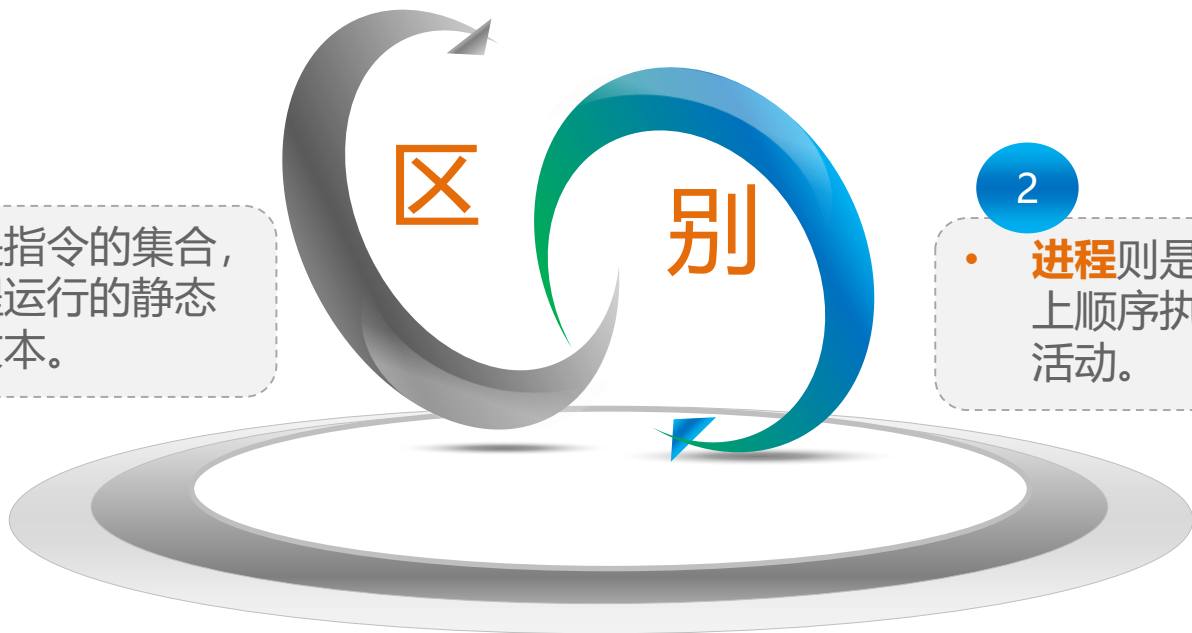
- **程序**是指令的集合，是进程运行的静态描述文本。

区

别

2

- **进程**则是程序在系统上顺序执行时的动态活动。



进程管理

进程的分类：

- 交互进程
- 批处理进程
- 守护进程



进程管理

进程的属性:

- 进程号 (PID)
- 父进程号 (PPID)
- 进程名
- 用户
- cpu%
- 内存%
- 优先级
- 开启时间等

进程								
资源								
文件系统								
进程名	用户	状态	虚拟内存	% CPU	CPU 时间	ID	内存	优先级
evolution-source-regis	root	睡眠	1005.0 MiB	0	0:00.18	1840	5.6 MiB	普通
nautilus	root	睡眠	999.1 MiB	0	0:03.93	1908	10.1 MiB	普通
tracker-extract	root	睡眠	894.4 MiB	0	0:01.25	1941	13.6 MiB	非常低
gnome-software	root	睡眠	741.7 MiB	0	0:00.39	1933	4.6 MiB	普通
tracker-miner-fs	root	睡眠	698.5 MiB	0	0:01.52	1950	3.2 MiB	非常低
goa-daemon	root	睡眠	697.3 MiB	0	0:00.16	1860	6.8 MiB	普通
libvirtd	root	睡眠	606.0 MiB	0	0:00.85	1059	572.0 KiB	普通
gnome-system-monito	root	运行	598.7 MiB	5	0:34.75	8912	16.8 MiB	普通
gsd-printer	root	睡眠	592.6 MiB	0	0:00.04	1804	2.5 MiB	普通
gnome-shell-calendar	root	睡眠	590.3 MiB	0	0:00.18	1831	3.9 MiB	普通
tracker-store	root	睡眠	589.6 MiB	0	0:01.01	1919	7.4 MiB	普通
caribou	root	睡眠	578.7 MiB	0	0:05.15	1849	3.9 MiB	普通
gnome-session	root	睡眠	564.1 MiB	0	0:00.53	1560	1.3 MiB	普通

进程管理

启动进程命令

启动前台进程

- 用户在shell中输入一条命令，即可启动一个前台进程，这是最常用的方式。
 - 例： `vim /proc/cpuinfo`

启动后台进程

- 只需在命令后面添加一个字符“&”
 - 例： `find / -name root >result.txt &`
- `jobs [-l]` 查看当前shell中已经启动的后台进程的执行状态。
- 将后台进程唤回前台执行： `fg %n`（n代表后台进程的工作号）

进程管理

查看进程状态

uptime

- 查看系统的当前时间、系统状态、用户已登录系统的时间、登录系统的用户数量
- 过去的1、5、15分钟内运行队列中的平均进程数量

ps

- 监控系统后台进程的工作情况
- `ps -ef`
- `ps -aux`

top

- 显示进程所占系统资源
- 动态查看系统中正在运行的进程的状态，默认每隔3秒刷新一次
- 在top程序的执行过程中输入命令，以交互方式控制执行结果
- `top -bn1` 一次性全部把所有信息输出出来而非动态显示

进程管理

终止进程命令

终止前台进程

- 在程序退出时，该前台进程将自动结束
- 在程序运行过程中，可按**Ctrl+c**键退出前台进程

终止后台进程

- 使用kill命令
 - **kill -15 pid** //正常结束一个进程
 - **kill -9 pid** //强制终止一个进程