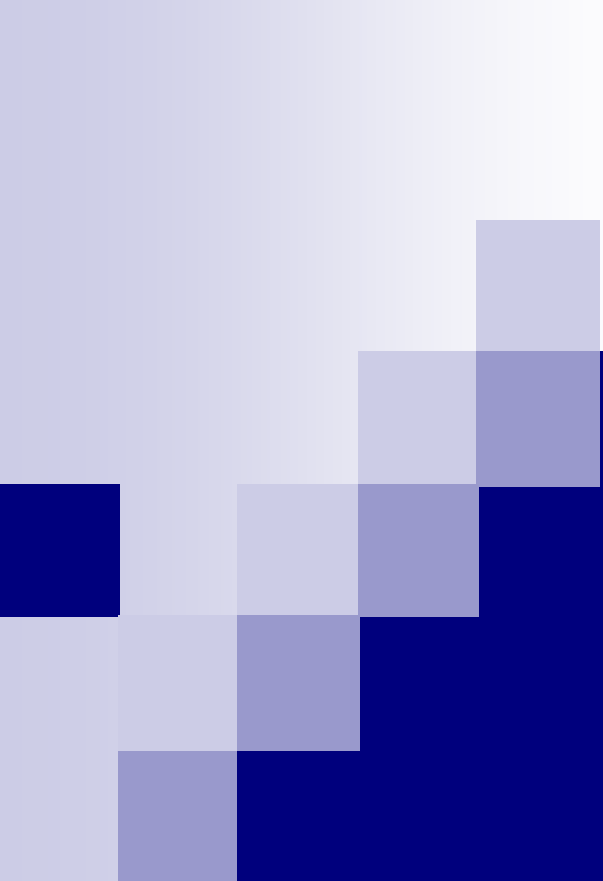




行程管理與排程

大綱

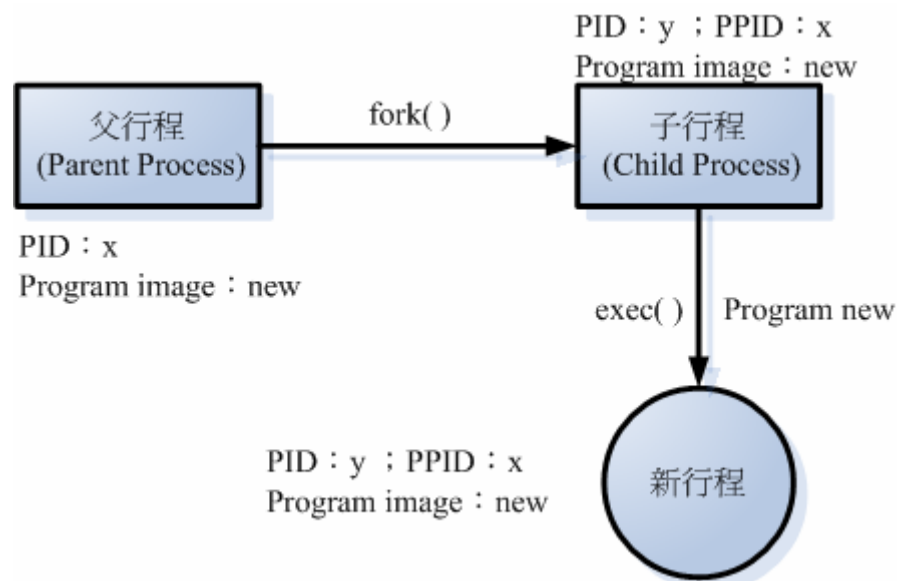
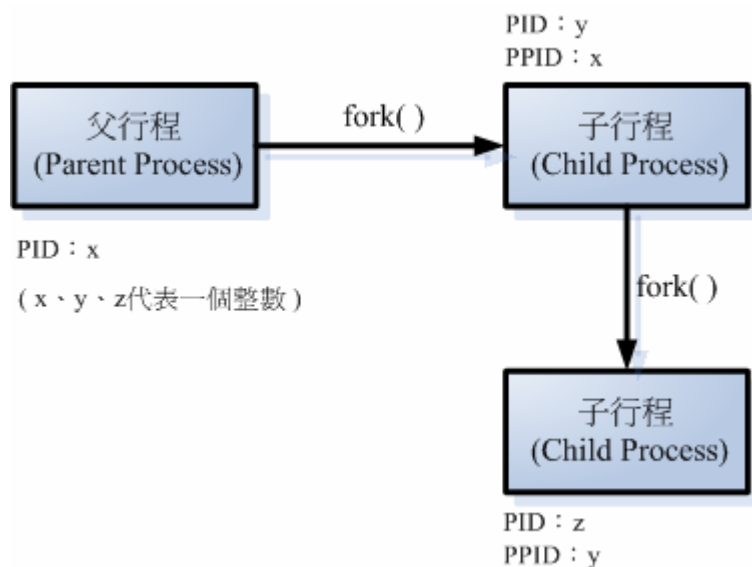
- 何謂行程
- 行程的產生
- 行程的終止
- 行程的狀態
- 執行緒
- 排程
- 環境切換
- Linux行程管理工具
- 本章重點回顧

何謂行程

- 作業系統中兩個非常抽象的觀念：檔案、行程
- 行程：簡單地說就是執行中的程式
- 包含許多資源，像是狀態信號、處理器狀態、記憶體位址空間，或是執行緒等。
- 傳統的Unix系統中，一個行程都只會包含一個執行緒

行程的產生

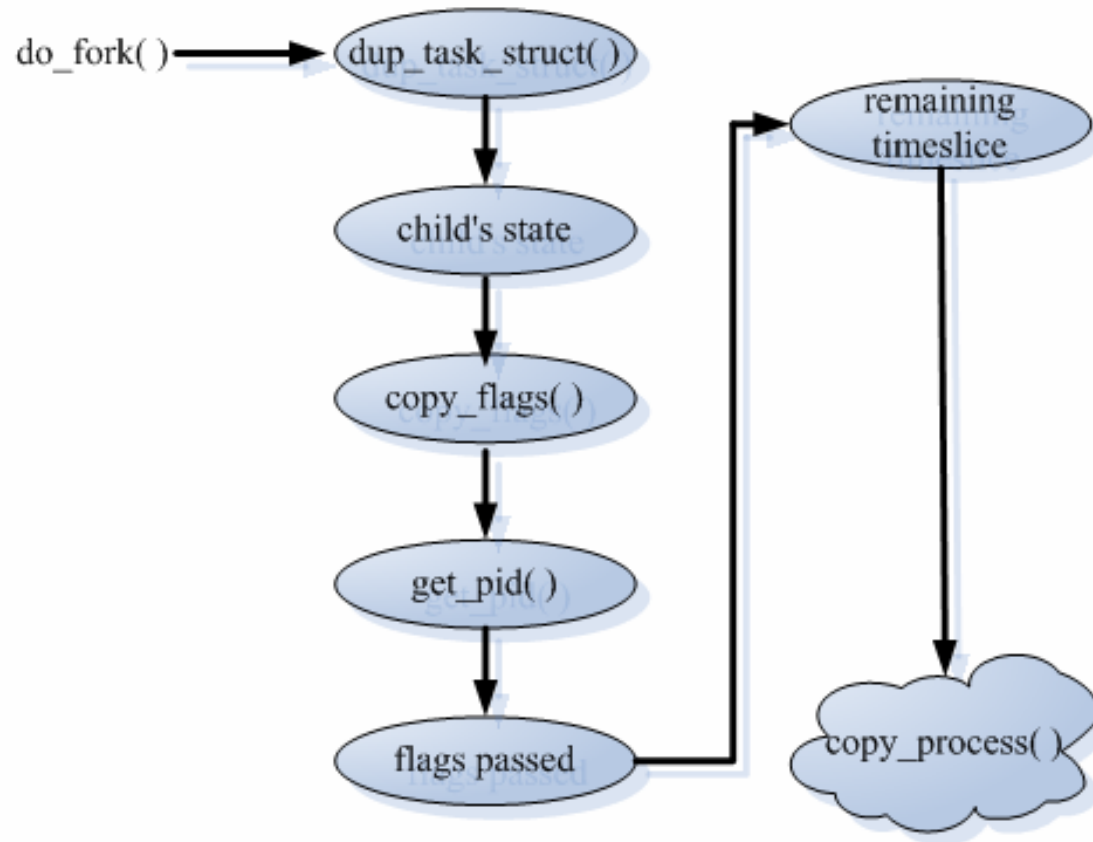
■ fork-and-exec運作模式：fork()和exec()



■ 在Linux核心中透過 `do_fork()` 的運作產生行程

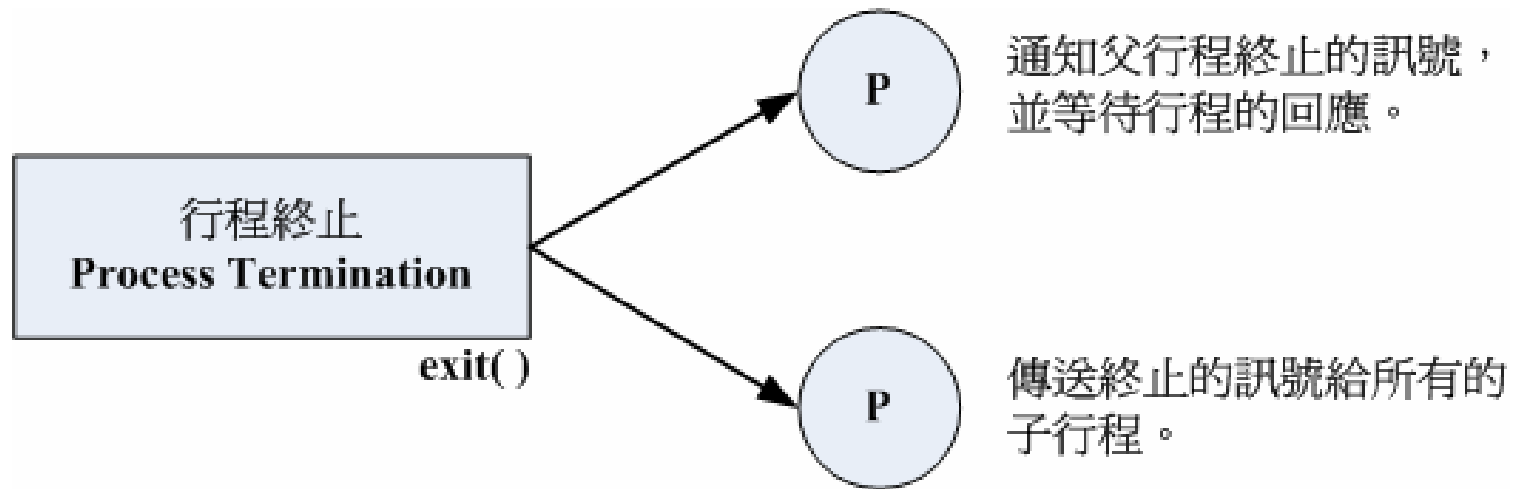
User Level

Kernel Level



行程的終止

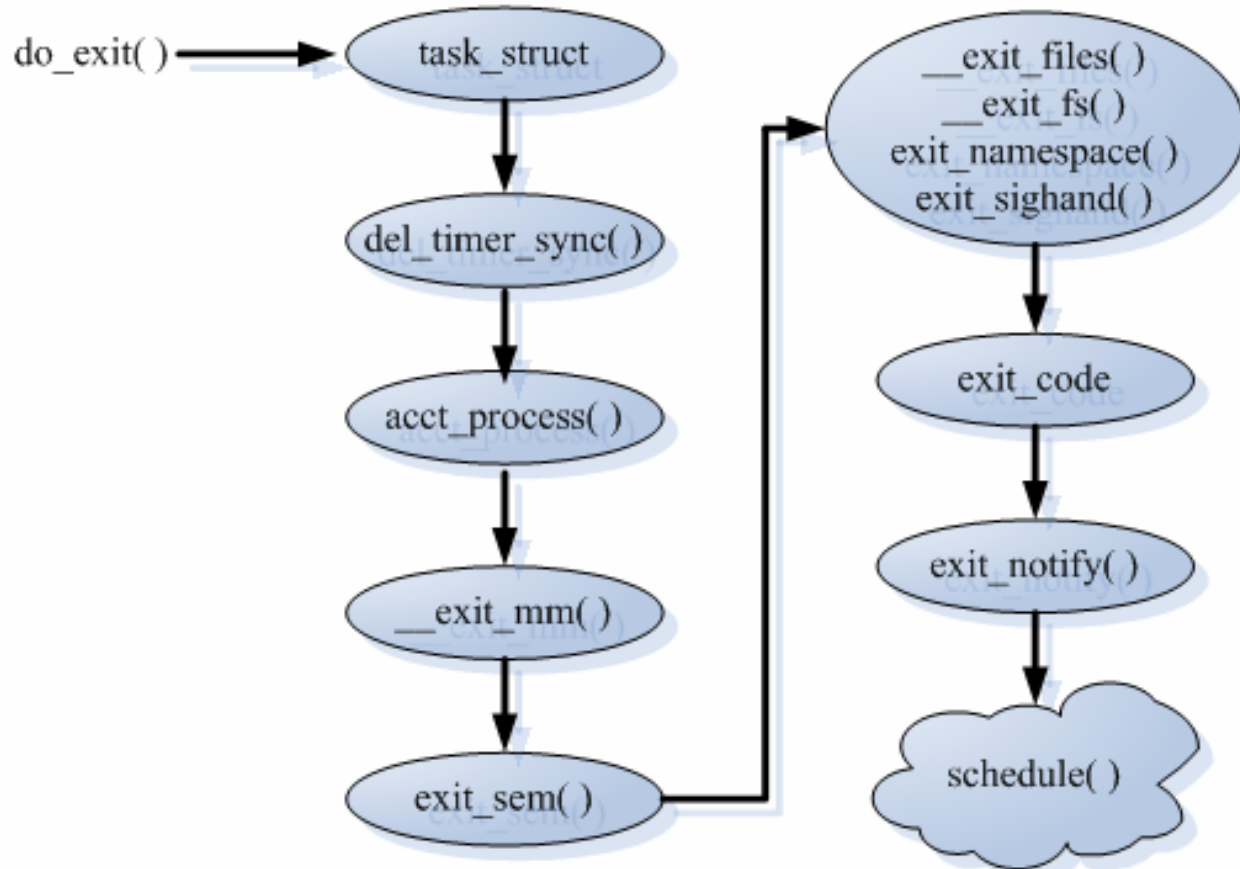
- 行程終止時，核心會釋放這個行程原先擁有的系統資源，並且送出一個訊息告訴所有的子行程，這個父行程被終止了。



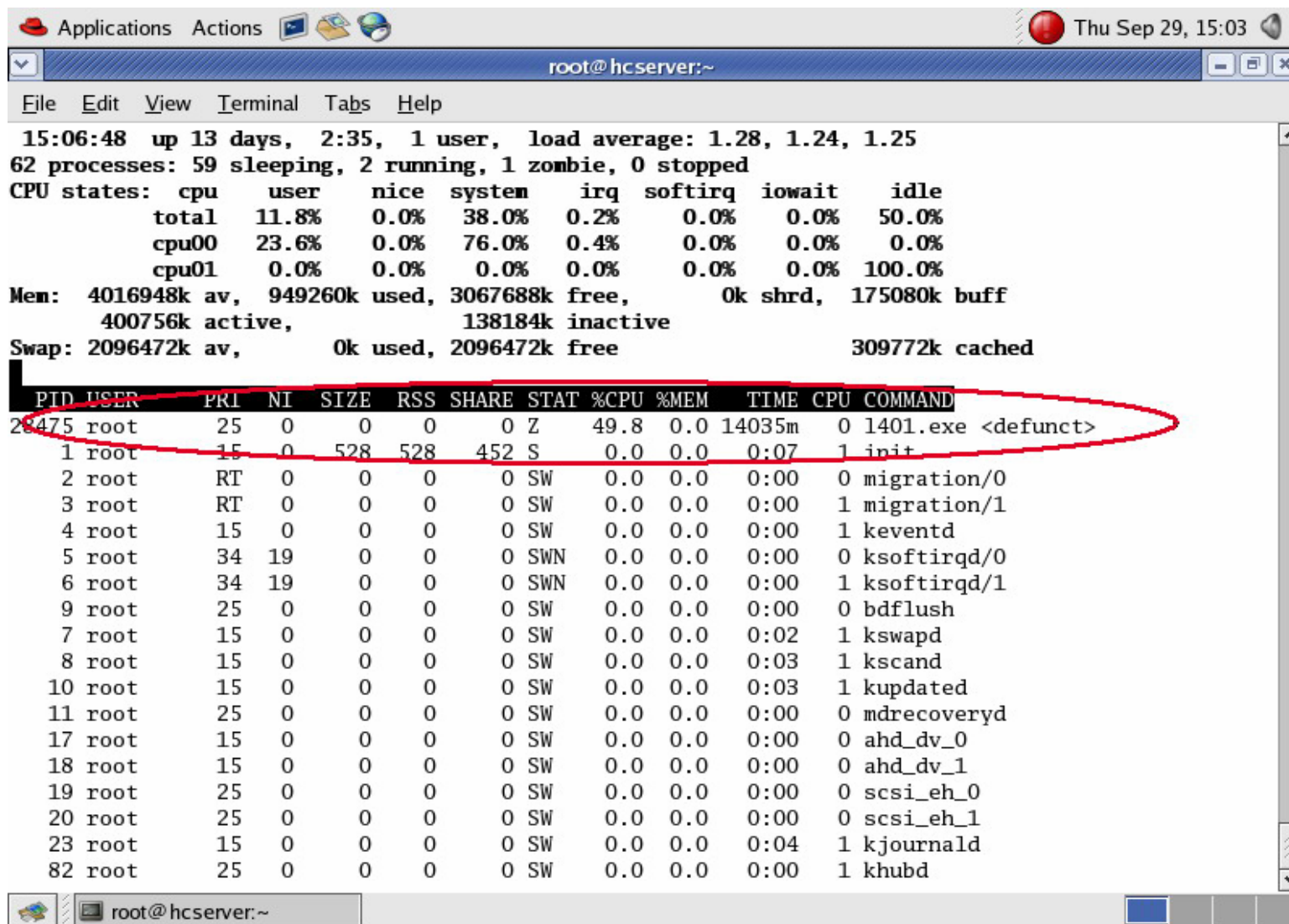
■ 在Linux核心中透過 do_exit()的運作終止行程

User Level

Kernel Level



■ Zombie Process/Ddefunct Process



Applications Actions Thu Sep 29, 15:03

root@hcserver:~

File Edit View Terminal Tabs Help

15:06:48 up 13 days, 2:35, 1 user, load average: 1.28, 1.24, 1.25
62 processes: 59 sleeping, 2 running, 1 zombie, 0 stopped

CPU states: cpu user nice system irq softirq iowait idle
total 11.8% 0.0% 38.0% 0.2% 0.0% 0.0% 50.0%
cpu00 23.6% 0.0% 76.0% 0.4% 0.0% 0.0% 0.0%
cpu01 0.0% 0.0% 0.0% 0.0% 0.0% 0.0% 100.0%

Mem: 4016948k av, 949260k used, 3067688k free, 0k shrd, 175080k buff
400756k active, 138184k inactive

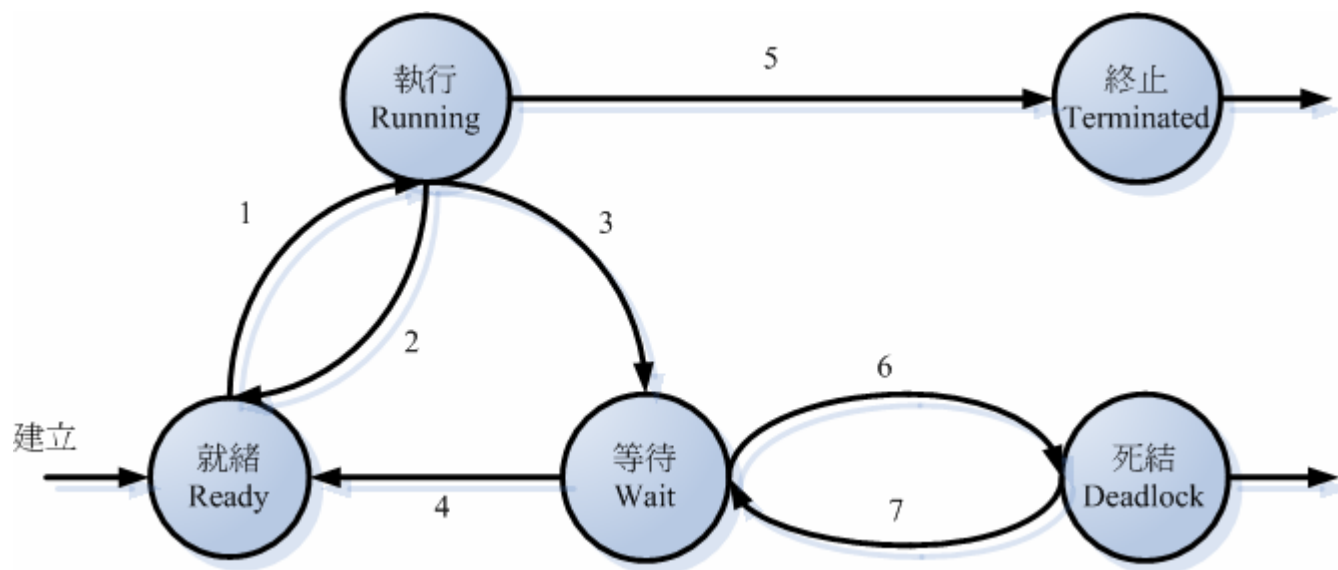
Swap: 2096472k av, 0k used, 2096472k free 309772k cached

PID	USER	PR	NI	SIZE	RSS	SHARE	STAT	%CPU	%MEM	TIME	CPU	COMMAND
28475	root	25	0	0	0	0	Z	49.8	0.0	14035m	0	1401.exe <defunct>
1	root	15	0	528	528	452	S	0.0	0.0	0:07	1	init
2	root	RT	0	0	0	0	SW	0.0	0.0	0:00	0	migration/0
3	root	RT	0	0	0	0	SW	0.0	0.0	0:00	1	migration/1
4	root	15	0	0	0	0	SW	0.0	0.0	0:00	1	keventd
5	root	34	19	0	0	0	SWN	0.0	0.0	0:00	0	ksoftirqd/0
6	root	34	19	0	0	0	SWN	0.0	0.0	0:00	1	ksoftirqd/1
9	root	25	0	0	0	0	SW	0.0	0.0	0:00	0	bdflood
7	root	15	0	0	0	0	SW	0.0	0.0	0:02	1	kswapd
8	root	15	0	0	0	0	SW	0.0	0.0	0:03	1	kscand
10	root	15	0	0	0	0	SW	0.0	0.0	0:03	1	kupdated
11	root	25	0	0	0	0	SW	0.0	0.0	0:00	0	mdrecoveryd
17	root	15	0	0	0	0	SW	0.0	0.0	0:00	0	ahd_dv_0
18	root	15	0	0	0	0	SW	0.0	0.0	0:00	0	ahd_dv_1
19	root	25	0	0	0	0	SW	0.0	0.0	0:00	0	scsi_eh_0
20	root	25	0	0	0	0	SW	0.0	0.0	0:00	0	scsi_eh_1
23	root	15	0	0	0	0	SW	0.0	0.0	0:04	1	kjournald
82	root	25	0	0	0	0	SW	0.0	0.0	0:00	1	khud

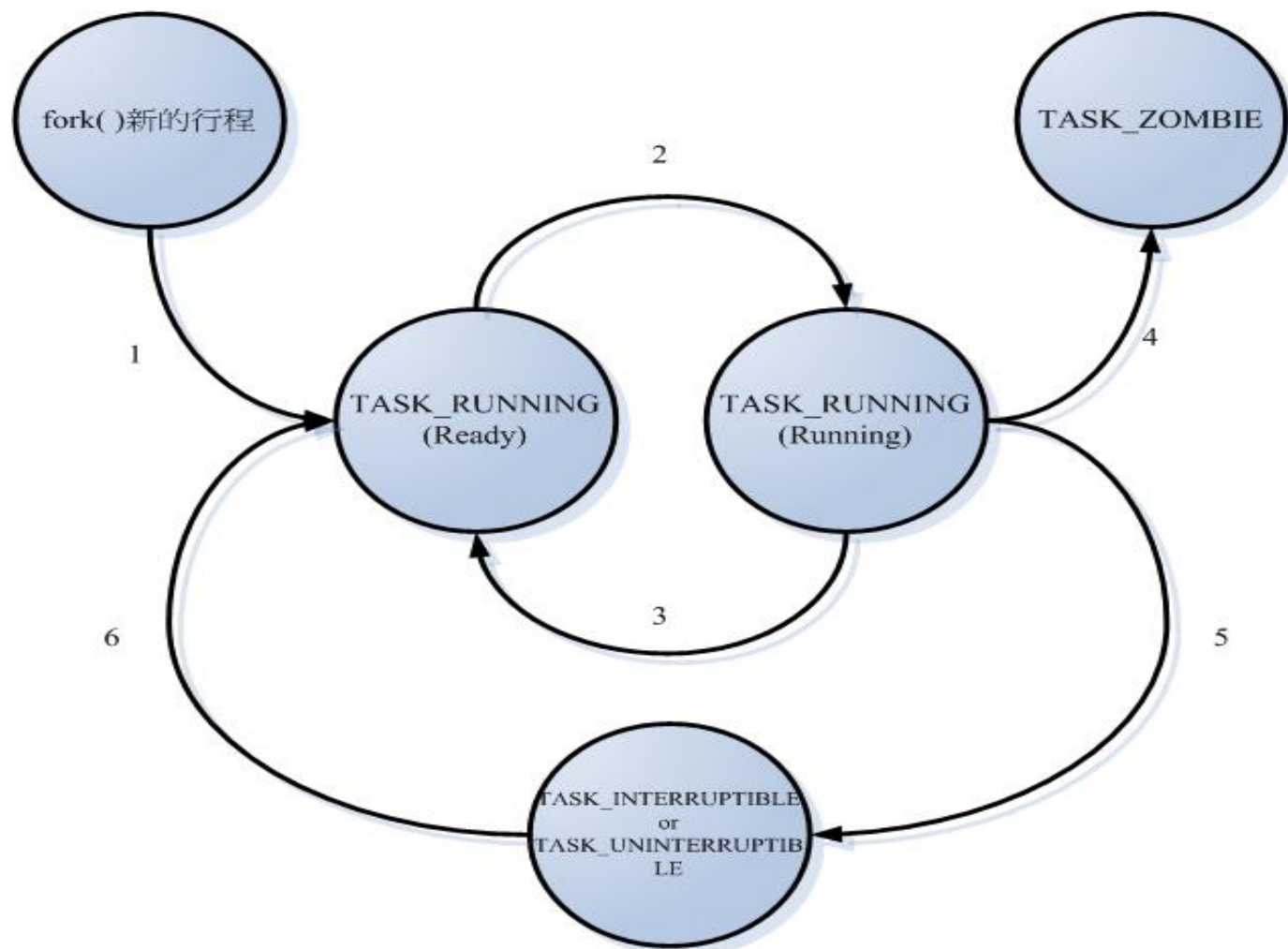
root@hcserver:~

行程的狀態

- 行程的狀態（State）：就緒（Ready）、執行（Running）、等待（Wait）、死結（Deadlock）和終止（Terminated）。

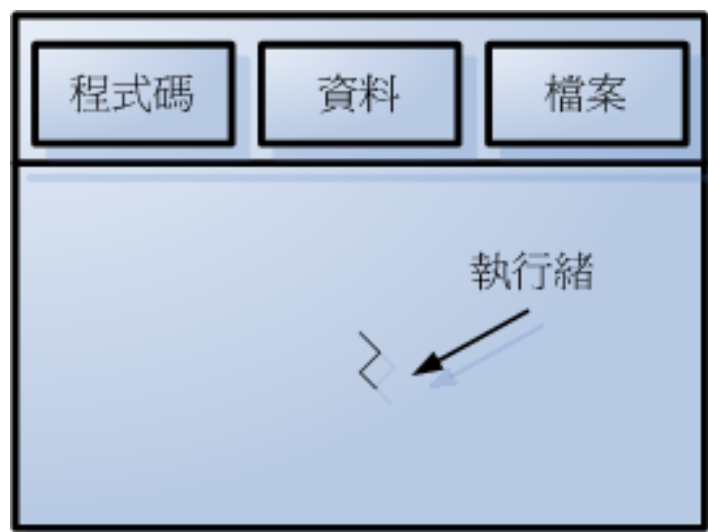


- Linux核心中針對這五種狀態定義了五個不同的旗標值

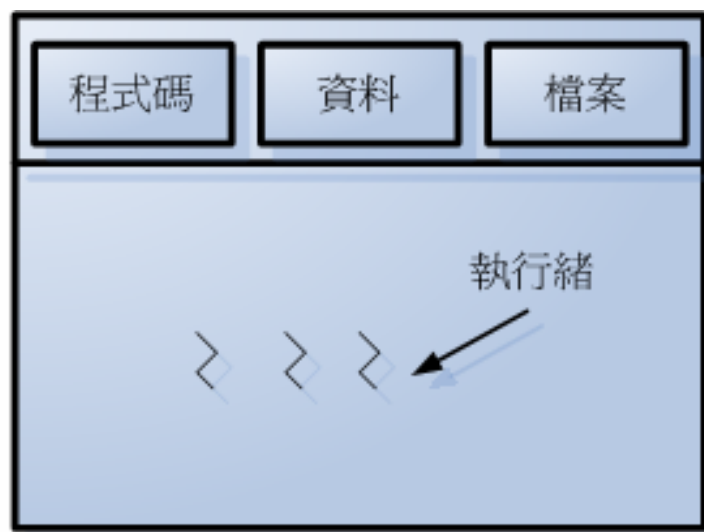


執行緒

- 執行緒（Threads）是CPU使用時的一個基本單位，裡面包含了一個執行緒ID、程式計數器、一組暫存器，以及一個堆疊空間所組成。

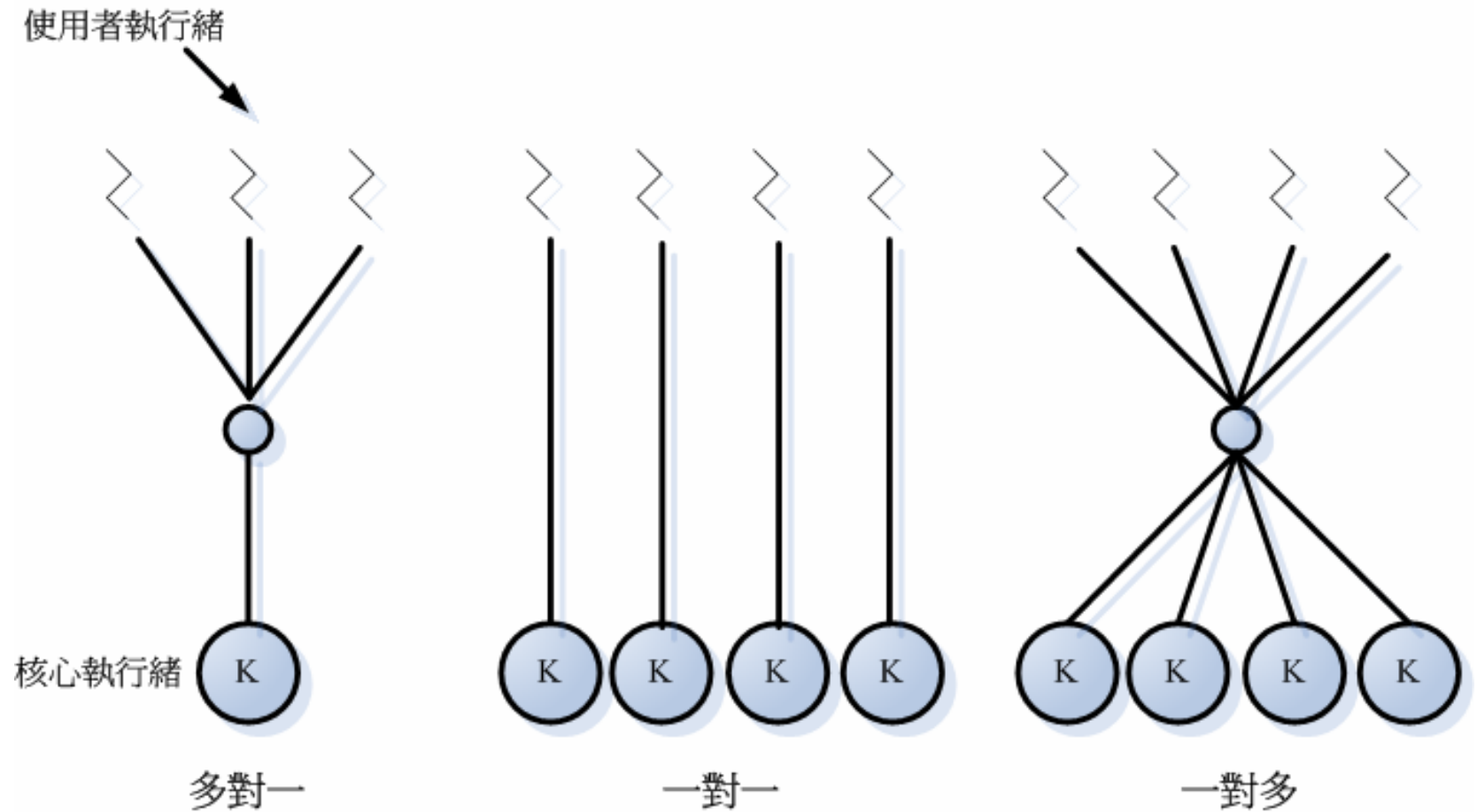


單執行緒行程



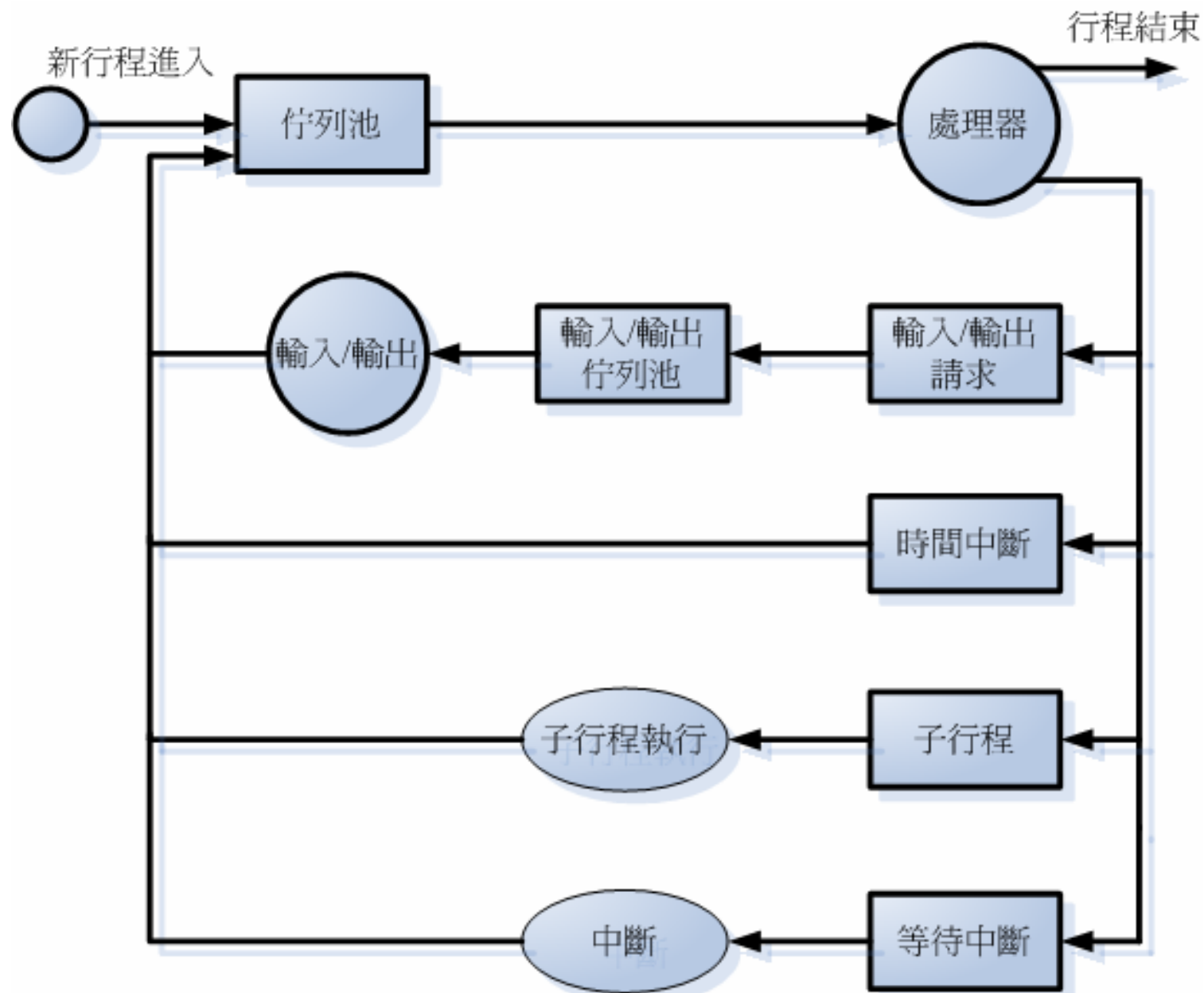
多執行緒行程

■ 使用者執行緒與核心執行緒



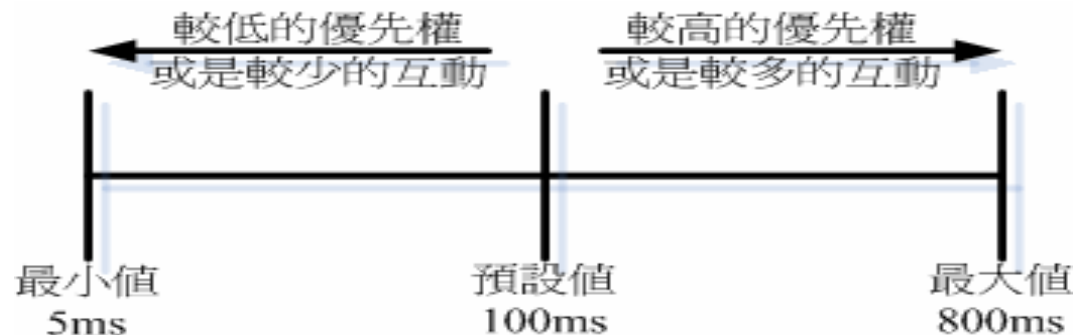
排程

- 排程（Scheduler）的政策會影響到整個系統CPU運用的效率
- 分類：輸入／輸出導向（I/O-Bound）和處理器導向（Processor-Bound）



- Linux作業系統中，利用優先權的大小來進行行程的排序，Linux排程程式會抽出時間分量值相對於較高的行程，並動態的決定執行的優先順序。

行程類別	Nice值	時間分量的值
最初狀態（產生）	父行程	父行程的一半
較小優先權	+19	5ms(MIN_TIMESLICE)
預設優先權	0	100ms(DEF_TIMESLICE)
較大優先權	-20	800ms(MAX_TIMESLICE)

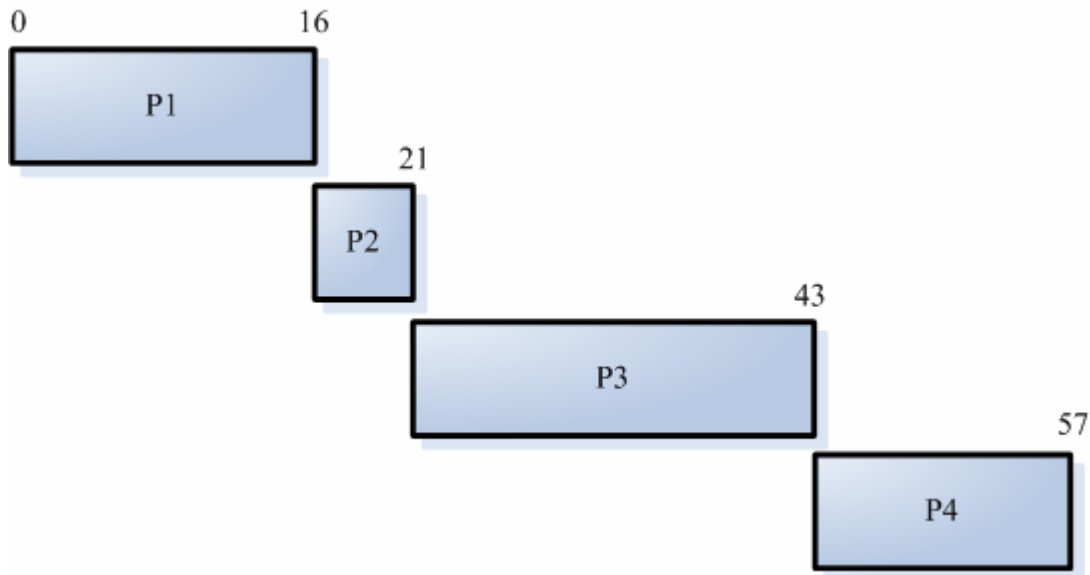


■ 在作業系統中常見的排程政策：

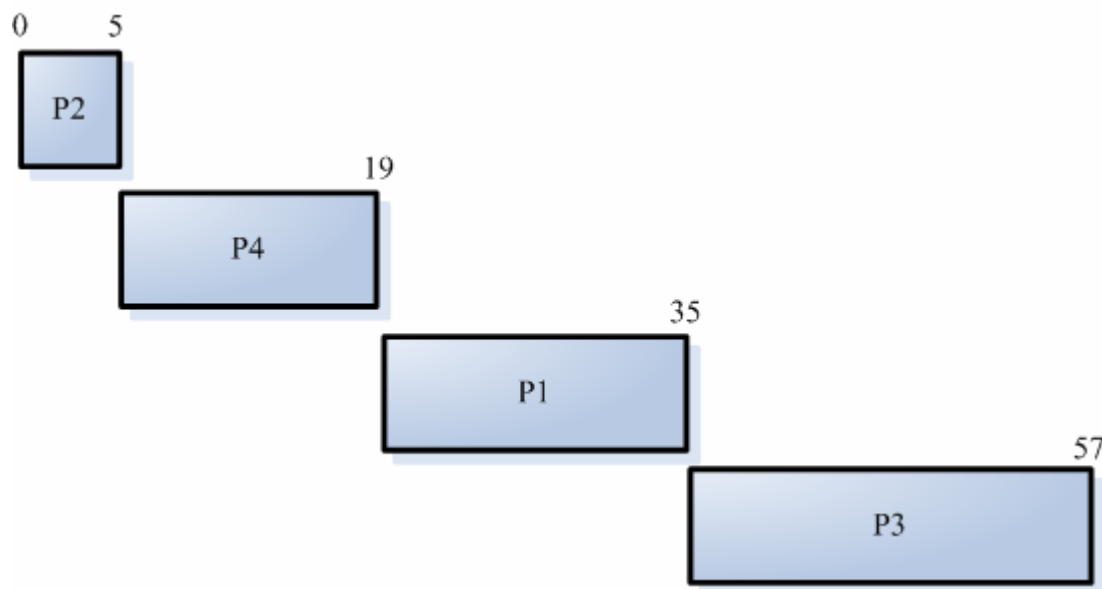
- 先來先服務的排程法（First Come First Served Scheduling，FCFS）
- 最短工作先排程法（Shortest Job First Scheduling，SJFS）
- 優先權排程法（Priority Scheduling）
- 依序循環工作排程法（Round-Robin Scheduling，RR）
- 多層佇列排程法（Multilevel Queue Scheduling）。

行程	時間分量（ms）	優先權
P1	16	3
P2	5	4
P3	22	2
P4	14	1

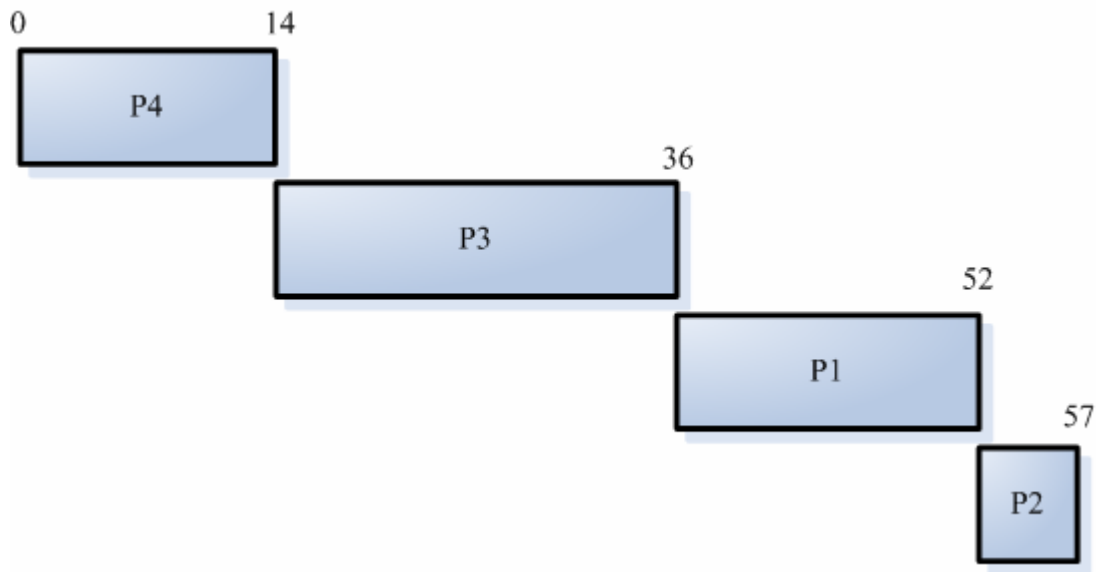
■ 先來先服務的排程法



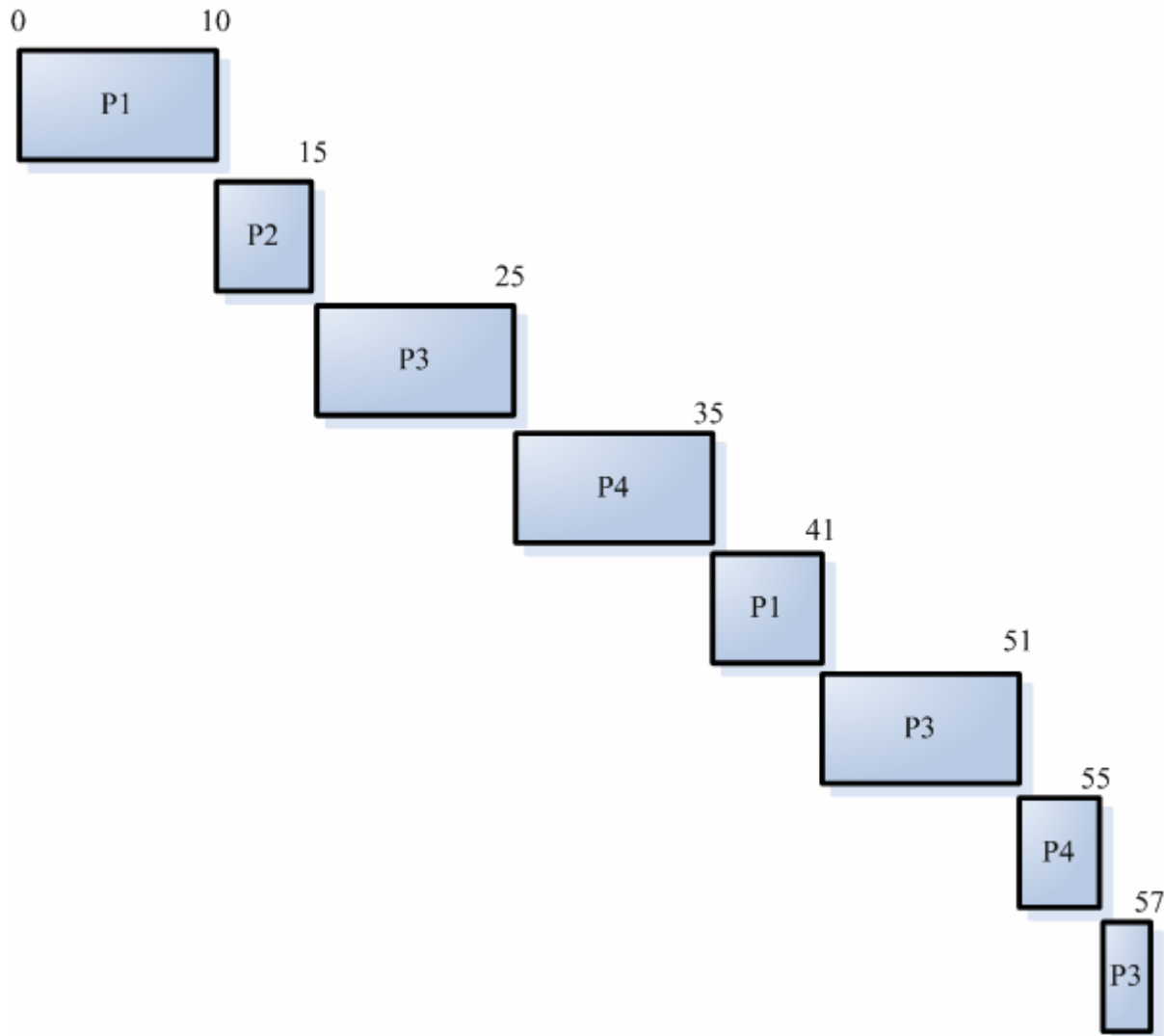
■ 最短工作先排程法



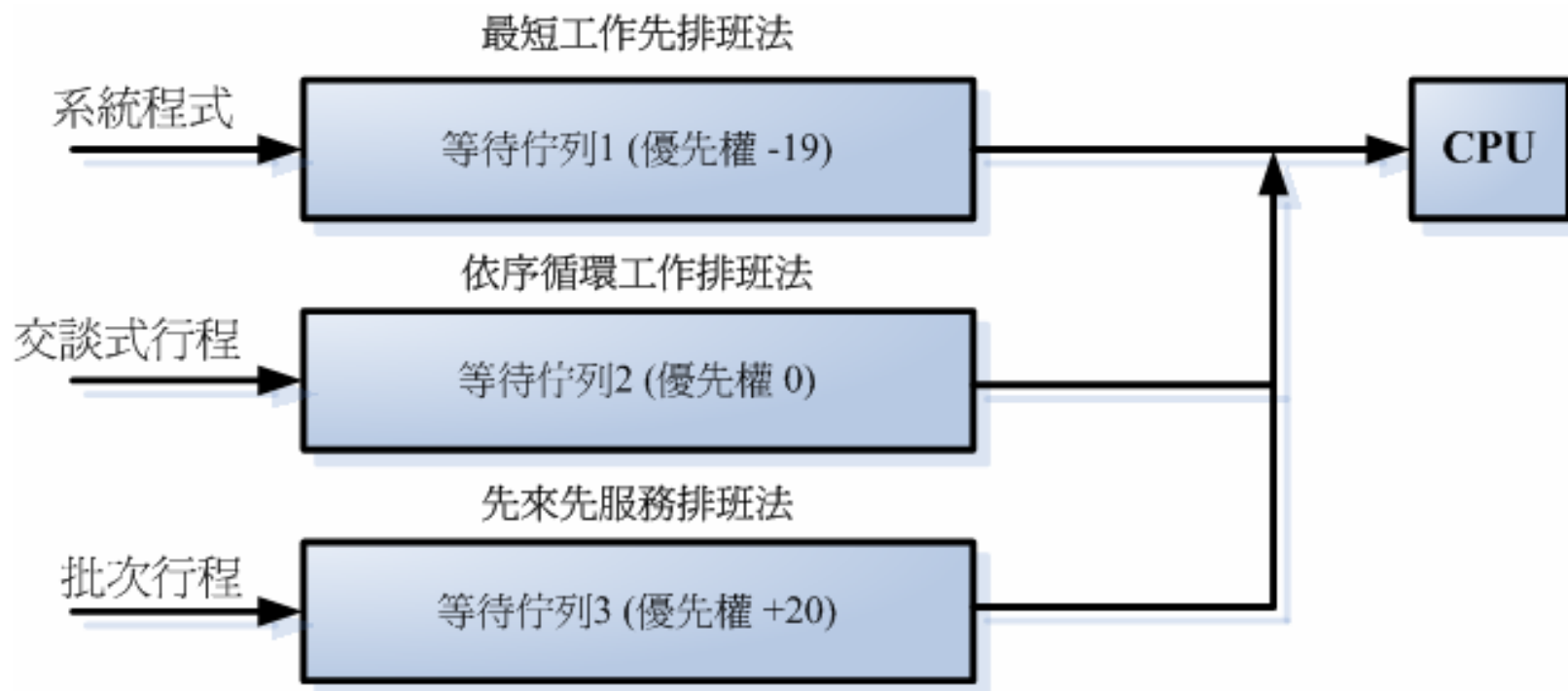
■ 優先權排程法



■ 依序循環工作排程法



■ 多層佇列排程法/多層回饋佇列排程法



■ 採用多層回饋佇列排程法有幾項特性：

- 系統不會發生飢餓（Starvation）現象
- 行程如果是屬於處理器導向者，會隨著時間而降低原先執行行程的優先順序，使用較小的時間分量。
- 行程如果是屬於輸入／輸出導向者，會隨著時間而提升原先執行行程的優先順序，使用較大的時間分量。



環境切換—就是監督程式！

- 環境切換（Context Switching，亦有人稱之為本文交換或是上下文交換）就是將一個正在執行中的工作行程執行權，透過context_switch()這個系統呼叫交給另外一個被排入執行的工作行程。
- $O(x)$ 排程架構：假設當行程數量越多時，所需要分配行程的時間也會成正比增加

Linux行程管理工具

■ ps

```
root@fedora5:~ - Shell - Konsole
工作階段 編輯 檢視 書籤 設定 說明

USER      PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root         1   0.1   0.1   1992   676 ?        S    15:11   0:01 init [5]

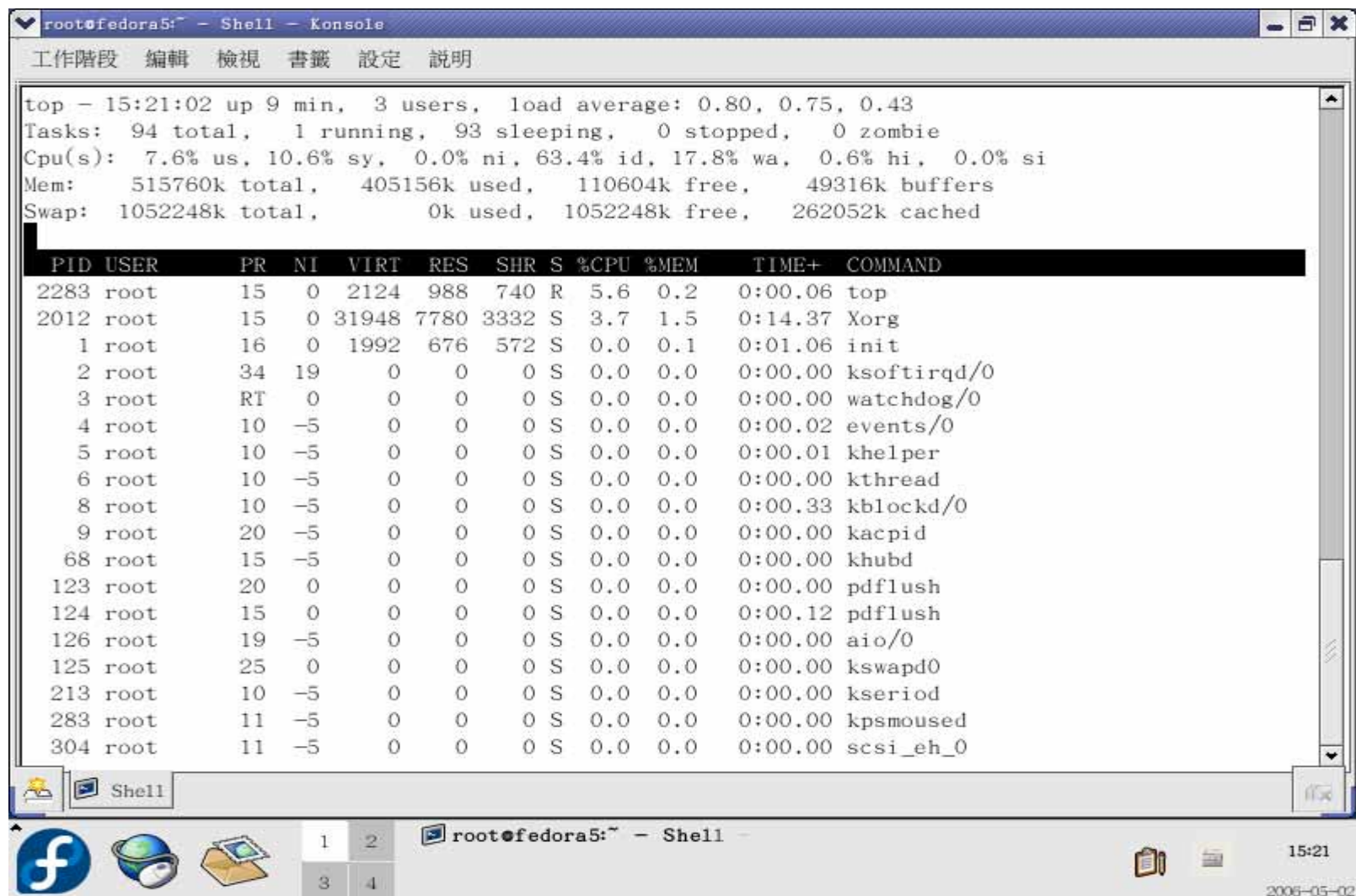
root        2   0.0   0.0     0     0 ?        SN   15:11   0:00 [ksoftirqd/0]
root        3   0.0   0.0     0     0 ?        S    15:11   0:00 [watchdog/0]
root        4   0.0   0.0     0     0 ?        S<   15:11   0:00 [events/0]
root        5   0.0   0.0     0     0 ?        S<   15:11   0:00 [khelper]
root        6   0.0   0.0     0     0 ?        S<   15:11   0:00 [kthread]
root        8   0.0   0.0     0     0 ?        S<   15:11   0:00 [kblockd/0]
root        9   0.0   0.0     0     0 ?        S<   15:11   0:00 [kacpid]
root       68   0.0   0.0     0     0 ?        S<   15:11   0:00 [khubd]
root      123   0.0   0.0     0     0 ?        S    15:11   0:00 [pdflush]
root      124   0.0   0.0     0     0 ?        S    15:11   0:00 [pdflush]
root      126   0.0   0.0     0     0 ?        S<   15:11   0:00 [aio/0]
root      125   0.0   0.0     0     0 ?        S    15:11   0:00 [kswapd0]
root      213   0.0   0.0     0     0 ?        S<   15:11   0:00 [kseriod]
root      283   0.0   0.0     0     0 ?        S<   15:11   0:00 [kpsmoused]
root      304   0.0   0.0     0     0 ?        S<   15:11   0:00 [scsi_eh_0]
root      321   0.0   0.0     0     0 ?        S<   15:11   0:00 [kmirrord]
root      332   0.1   0.0     0     0 ?        S    15:11   0:00 [kjournald]
root      370   0.0   0.0     0     0 ?        S<   15:11   0:00 [kauditd]
root      394   0.1   0.1   2208   676 ?        S< s 15:11   0:00 /sbin/udev -d
root      604   0.0   0.0     0     0 ?        S<   15:11   0:00 [kgameportd]
root     1080   0.0   0.0     0     0 ?        S    15:12   0:00 [kjournald]

--More--
```

Shell

1 2 3 4 root@fedora5:~ - Shell 15:20 2006-05-02

■ top

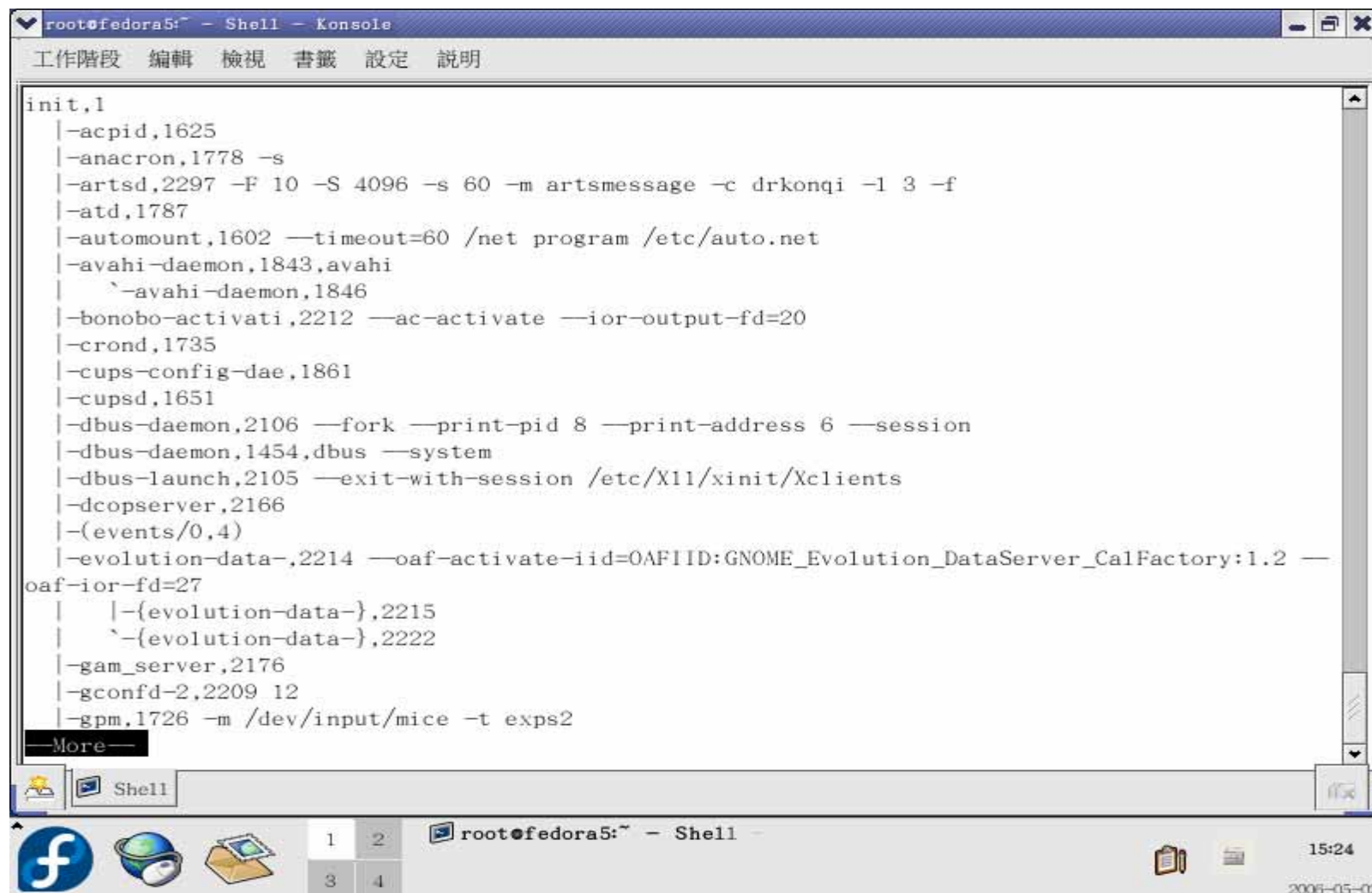


The screenshot shows a Fedora 5 desktop environment. A terminal window titled 'root@fedora5:~ - Shell - Konsole' is open, displaying the output of the 'top' command. The terminal window has a menu bar with '工作階段', '編輯', '檢視', '書籤', '設定', and '說明'. The 'top' command output shows system statistics and a list of running processes. The desktop has a taskbar at the bottom with icons for a file manager, a web browser, and a terminal. The system clock shows 15:21 on 2006-05-02.

```
top - 15:21:02 up 9 min, 3 users, load average: 0.80, 0.75, 0.43
Tasks: 94 total, 1 running, 93 sleeping, 0 stopped, 0 zombie
Cpu(s): 7.6% us, 10.6% sy, 0.0% ni, 63.4% id, 17.8% wa, 0.6% hi, 0.0% si
Mem: 515760k total, 405156k used, 110604k free, 49316k buffers
Swap: 1052248k total, 0k used, 1052248k free, 262052k cached
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
2283	root	15	0	2124	988	740	R	5.6	0.2	0:00.06	top
2012	root	15	0	31948	7780	3332	S	3.7	1.5	0:14.37	Xorg
1	root	16	0	1992	676	572	S	0.0	0.1	0:01.06	init
2	root	34	19	0	0	0	S	0.0	0.0	0:00.00	ksoftirqd/0
3	root	RT	0	0	0	0	S	0.0	0.0	0:00.00	watchdog/0
4	root	10	-5	0	0	0	S	0.0	0.0	0:00.02	events/0
5	root	10	-5	0	0	0	S	0.0	0.0	0:00.01	khelper
6	root	10	-5	0	0	0	S	0.0	0.0	0:00.00	kthread
8	root	10	-5	0	0	0	S	0.0	0.0	0:00.33	kblockd/0
9	root	20	-5	0	0	0	S	0.0	0.0	0:00.00	kacpid
68	root	15	-5	0	0	0	S	0.0	0.0	0:00.00	khubd
123	root	20	0	0	0	0	S	0.0	0.0	0:00.00	pdflush
124	root	15	0	0	0	0	S	0.0	0.0	0:00.12	pdflush
126	root	19	-5	0	0	0	S	0.0	0.0	0:00.00	aio/0
125	root	25	0	0	0	0	S	0.0	0.0	0:00.00	kswapd0
213	root	10	-5	0	0	0	S	0.0	0.0	0:00.00	kseriod
283	root	11	-5	0	0	0	S	0.0	0.0	0:00.00	kpsmoused
304	root	11	-5	0	0	0	S	0.0	0.0	0:00.00	scsi_eh_0

■ pstree



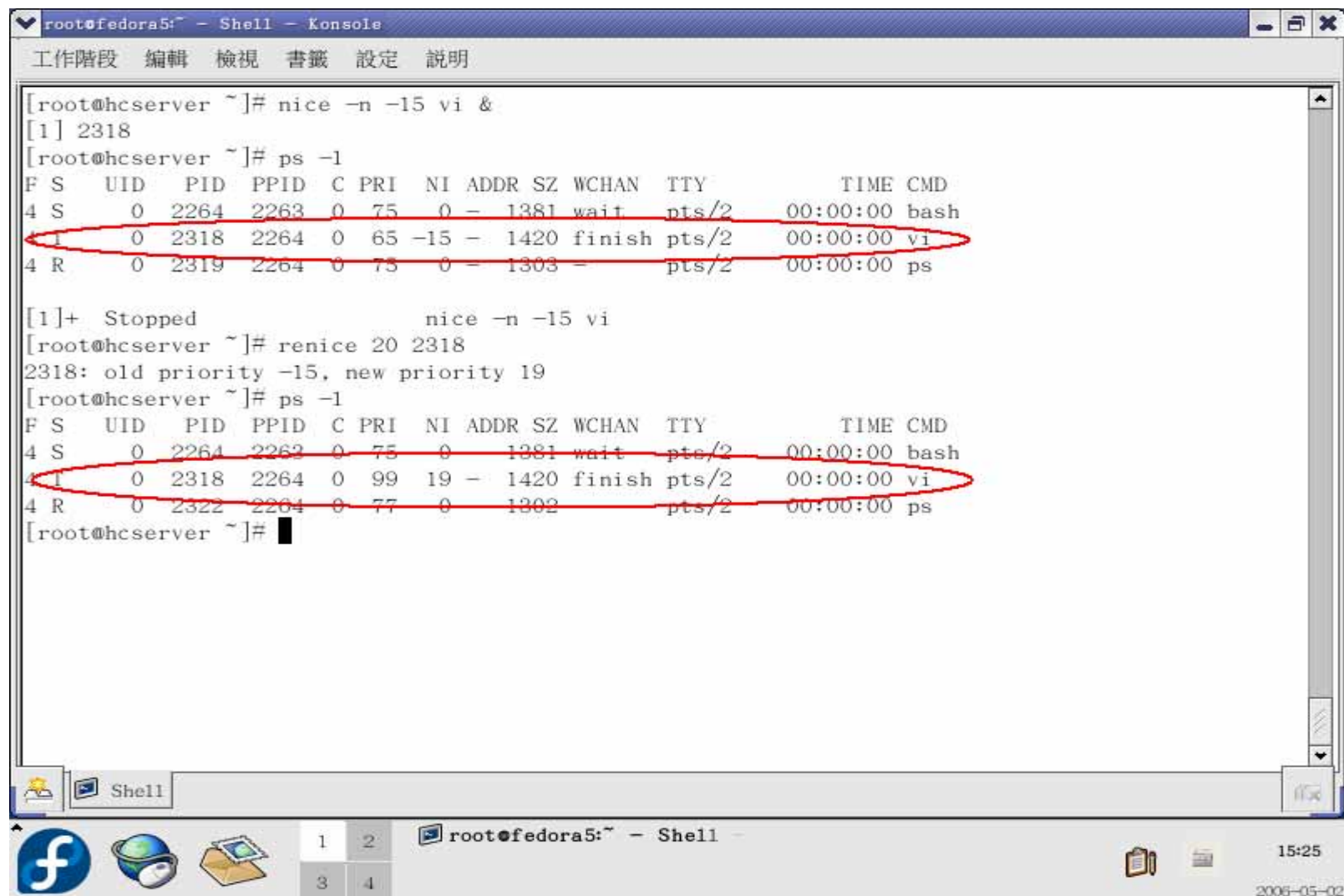
```
root@fedora5:~ - Shell - Konsole
工作階段 編輯 檢視 書籤 設定 說明

init,1
|-acpid,1625
|-anacron,1778 -s
|-artsd,2297 -F 10 -S 4096 -s 60 -m artsmessage -c drkonqi -l 3 -f
|-atd,1787
|-automount,1602 --timeout=60 /net program /etc/auto.net
|-avahi-daemon,1843,avahi
|   |-avahi-daemon,1846
|-bonobo-activati,2212 --ac-activate --ior-output-fd=20
|-crond,1735
|-cups-config-dae,1861
|-cupsd,1651
|-dbus-daemon,2106 --fork --print-pid 8 --print-address 6 --session
|-dbus-daemon,1454,dbus --system
|-dbus-launch,2105 --exit-with-session /etc/X11/xinit/Xclients
|-dcopserver,2166
|-(events/0,4)
|-evolution-data-,2214 --oaf-activate-iid=OAFIID:GNOME_Evolution_DataServer_CalFactory:1.2 --
oaf-ior-fd=27
|   |-(evolution-data-),2215
|   |   |-{evolution-data-},2222
|-gam_server,2176
|-gconfd-2,2209 12
|-gpm,1726 -m /dev/input/mice -t exps2
--More--

Shell

1 2
3 4 root@fedora5:~ - Shell
15:24
2006-05-02
```

■ nice/renice



The screenshot shows a terminal window titled "root@fedora5:~ - Shell - Konsole". The user runs the command `nice -n -15 vi &`, which starts a `vi` editor process with a priority of -15. The process ID is 2318. The user then runs `ps -l` to list the processes. The output shows three processes: `bash` (PID 2264, priority 75), `vi` (PID 2318, priority -15), and `ps` (PID 2319, priority 75). The `vi` process is circled in red. The user then presses `[1]+` to stop the `nice` command and runs `renice 20 2318` to change the priority of the `vi` process to 20. The output shows the old priority was -15 and the new priority is 19. The user runs `ps -l` again, and the output shows the `vi` process now has a priority of 19. The `vi` process is again circled in red.

```
[root@hcserver ~]# nice -n -15 vi &
[1] 2318
[root@hcserver ~]# ps -l
F S  UID  PID  PPID  C PRI  NI ADDR SZ WCHAN  TTY      TIME CMD
4 S   0  2264  2263  0  75   0 -  1381 wait  pts/2    00:00:00 bash
4 T   0  2318  2264  0  65 -15 -  1420 finish pts/2    00:00:00 vi
4 R   0  2319  2264  0  75   0 -  1303 -    pts/2    00:00:00 ps

[1]+  Stopped                  nice -n -15 vi
[root@hcserver ~]# renice 20 2318
2318: old priority -15, new priority 19
[root@hcserver ~]# ps -l
F S  UID  PID  PPID  C PRI  NI ADDR SZ WCHAN  TTY      TIME CMD
4 S   0  2264  2263  0  75   0 -  1381 wait  pts/2    00:00:00 bash
4 T   0  2318  2264  0  99  19 -  1420 finish pts/2    00:00:00 vi
4 R   0  2322  2264  0  77   0 -  1302 -    pts/2    00:00:00 ps
[root@hcserver ~]#
```

本章重點回顧

- 了解何謂行程與行程產生與終止的方式。
- 了解行程狀態所代表的意義。
- 了解何謂執行緒。
- 了解不同行程排程演算法所帶來的影響。
- 了解於Linux作業系統中管理行程的狀態。

