

WUOLAH



Alberto188

www.wuolah.com/student/Alberto188



191

resument 4-6.pdf

PAS-ResumenTeoria



2º Programación y Administración de Sistemas



Grado en Ingeniería Informática



**Escuela Politécnica Superior de Córdoba
UCO - Universidad de Córdoba**

 **escuela
de negocios**
CÁMARA DE SEVILLA

MÁSTER EN DIRECCIÓN Y GESTIÓN DE RECURSOS HUMANOS

www.mastersevilla.com

Inscríbete



BECAS

Tema 4: Gestión de usuarios

- Definición de usuario:

1. Persona que trabaja en el sistema: editando ficheros, ejecutando programas ...
2. Pseudo-usuario: entidad, que sin ser una persona, puede ejecutar programas o poseer ficheros.

Como características posee un nombre de usuario (logname o username), el identificador del usuario (ya que el sistema trabaja directamente con este UID) y los identificadores de los grupos a los que pertenece (GIDS).

- Principales ficheros de configuración:

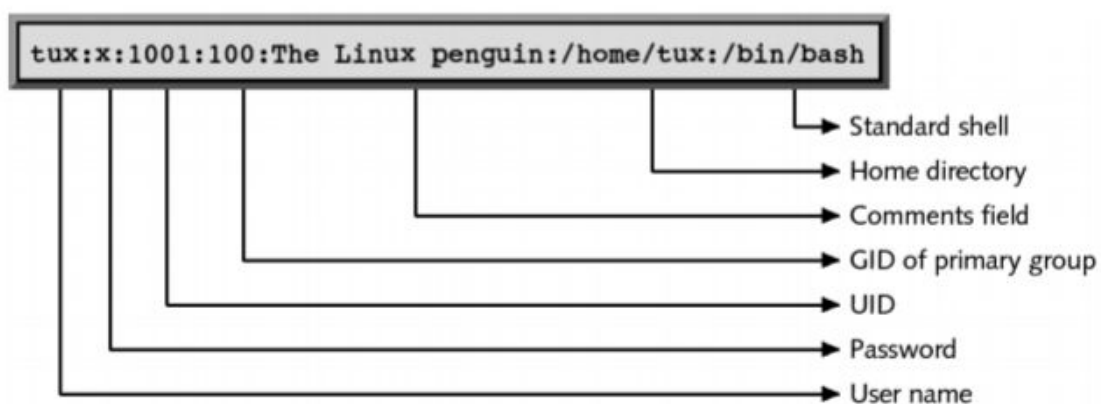
1. `/etc/passwd` -> información de las cuentas de usuarios.
2. `/etc/shadow` -> passwords cifradas e información de envejecimiento de las cuentas.
3. `/etc/group` -> definición de los grupos y usuarios miembros.
4. `/etc/gshadow` -> passwords de grupos cifradas.

- Fichero `/etc/passwd`: contiene la lista de usuarios y contraseñas. El formato que posee es el siguiente:

nombre:password:uid:gid:gecos:home:shell

- gecos → campo de información referente al usuario (nombre, teléfono, ...).
- shell → Intérprete de órdenes

1. El propietario del fichero es root y el grupo root.
2. Se permite el acceso al fichero `etc/passwd` en modo lectura para poder leer información del usuario, pero no se debería permitir acceso a las passwords.





Striking Vipers - black mirror



Toy Story 4



Segunda temporada de Dark

Próximos estrenos de series y películas.

Se acercan las vacaciones de verano y es el mejor momento de disfrutar de la televisión.

Las vacaciones ya están aquí. Si no has tenido mucho tiempo libre para ver cine y televisión ahora es tu momento. Muchos han sido los estrenos este 2019 que han movido las masas de fans. Sin embargo, las puertas al verano no van a dejar indiferente al espectador.

Si durante el curso no has visto nada desde el verano pasado, siempre puedes continuar todas la series acumuladas. Pero por si no estás al día, aquí os dejamos algunos estrenos próximos:

- **Aladdín.** Disney no nos dejará indiferente con la versión real de Aladdín. Esta historia de amor que todos conocemos ha saltado a la gran pantalla el 24 de mayo, y aún tienes tiempo para ir a verla. El dato más fascinante de este estreno es que el mago más pícaro de todo cuento está interpretado por Will Smith.

- **Black Mirror.** Desde el estreno de su película interactiva Bandersnatch en diciembre de 2018, Black Mirror se ha mantenido ausente hasta el estreno de la quinta temporada. La serie de antología que saca el lado más oscuro de la tecnología actual, ha dado paso con su última temporada a tres nuevos episodios el pasado 5 de junio.

Aladdín entre las películas de estreno más taquilleras.

- **El cuento de la criada.** Esta serie dramática basada en el premiado best-seller de Margaret Atwood, que muestra la lucha contra la antigua sociedad totalitaria de Estados Unidos, ha lanzado su tercera temporada también el 5 de junio.

- **X-Men: Dark Phoenix.** Basada en el personaje de Fénix Oscura, esta nueva entrega de los X-Men dió el salto a la gran pantalla el pasado viernes 7 de junio. Un dato relevante es que su protagonista está interpretada por Sophie Turner, que encarna el personaje de Sansa en Juego de Tronos.

- **Jessica Jones.** Volvemos con Marvel, esta vez en la televisión. La tercera y última temporada de la superheroína y detective Jessica Jones, ambientada en el Universo Cinematográfico de Marvel (UCM), se estrenará el próximo 14 de junio.

- **Toy Story 4.** Si eras un niño cuando se estrenó por primera vez Toy Story estamos seguros de que esta saga es especial para ti. La saga de animación sobre los juguetes más rebeldes no deja indiferente a ningún niño, joven o adulto. Por ello, Toy Story llega con más fuerza que nunca con su cuarta película, que tiene previsto su estreno el 21 de junio.

- **Stranger things.** La serie de ciencia ficción que ha dado vida al mitológico Demogorgon y protagonizada por Winona Ryder entre muchos otros actores, ha

La Casa de Papel, éxito mundial en numerosos países.

vuelto con su tercera temporada. Después de 2 temporadas de puro éxito, la serie vuelve a la pantalla el 4 de julio.

- **Spider-man: Far From Home.** Después del ansiado desenlace de Vengadores: Endgame llega Spider-man: Lejos de Casa. La segunda parte de Spider-man: Homecoming se estrenará el próximo 5 de julio. ¿Cómo se encuentra Peter Parker después de lo ocurrido? ¿A qué nuevos retos deberá enfrentarse?

- **La Casa de Papel.** Los delincuentes más gamberro ahora tienen competencia: la inspectora Sierra. La tercera temporada tiene previsto su estreno el próximo 19 de julio. ¿Quién ganará esta batalla? ¿la inspectora Alicia Sierra o la aclamada resistencia?

- **El Rey León.** Sin duda, el largometraje con el que Disney quiere despertar nuestro sentimiento más profundo es El Rey León. También próximamente estrenada el 19 de julio, esta película sobre el fantástico reino animal y la vida del León Simba nos dejará con la boca abierta. Una maravilla obligatoria.

- **Dark.** La segunda temporada de esta webserie de tipo drama sobrenatural se estrenará el 21 de junio. A tan sólo 1 semana y 3 días para conocer más sobre los oscuros secretos que guardan estas cuatro familias alemanas que ponen de manifiesto el pasado de una ciudad.

Wuolah Giveaway

Impresora y escáner Canon. Es momento de imprimir tus apuntes. Participa y llévate esta impresora con escáner.

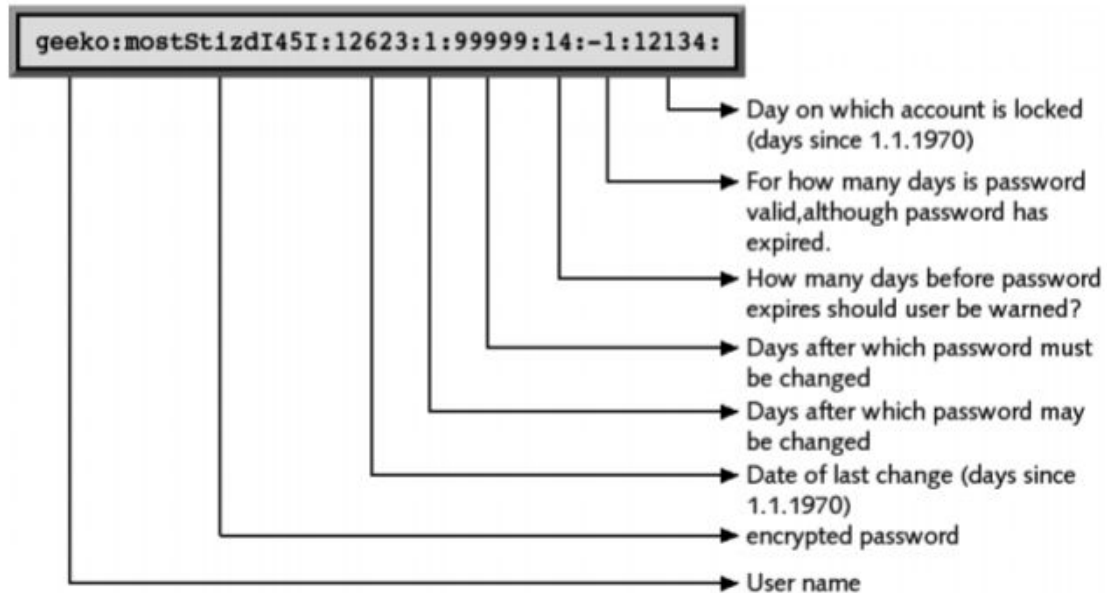


Wuolah Giveaway

Altavoz inalámbrico ZoeeTree. Siente la música allá donde vayas con este altavoz Bluetooth inalámbrico.

- Término: Shadow Passwords, permite que las contraseñas se guarden en un fichero más restringido que `etc/passwd`. Sólo permite la lectura y escritura por parte del usuario `root`, en algunos casos el usuario puede comprobar la contraseña.

- En el fichero se guarda para cada usuario del sistema, la contraseña cifrada junto con su información de envejecimiento.



- Tendría el siguiente formato:
nom:pass:changed:minlife:maxlife:warn:inactive:expired:unused

- Para crear una contraseña cifrada se valen del siguiente comando:
`mkpasswd --method=sha-512 contraseña salt`

- Se valen de dos algoritmos de bash para el cifrado de contraseñas:

1. MD5 (Message-Digest algorithm 5): Aplica funciones no lineales a los 17 segmentos de 32 bits de un bloque de 512 bits. Se obtiene un resumen de 128 bits.
2. SHA (Secure Hash Algorithm): Parecida a MD4, pero genera resúmenes más grandes, que lo hacen más seguro ante diferentes tipos de ataques. Se pueden considerar 160, 224, 256, 384 o 512 bits para el resumen. El resumen con 512 bits, es el usado de manera estándar por GNU/Linux.

-. Fichero `/etc/shadow`: busca introducir restricciones de tiempo o envejecimiento para la validez de la cuenta o contraseña.

- *changed* ⇒ fecha del último cambio de contraseña.
 - *minlife* ⇒ nº de días que han de pasar para poder cambiar la contraseña.
 - *maxlife* ⇒ nº de días máximo que puede estar con la misma contraseña sin cambiarla.
 - *warn* ⇒ cuántos días antes de que la contraseña expire (*maxlife*) el usuario será informado sobre ello, indicándole que tiene que cambiarla.
 - *inactive* ⇒ nº de días después de que la contraseña expire en que la cuenta se deshabilitará si no ha sido cambiada.
 - *expired* ⇒ fecha en la que la cuenta expira y se deshabilita de forma automática.
- Ejemplo de código para una contraseña:

```
1 chage -M 20 pagutierrez
2 chage -W 6 pagutierrez
3 chage -I 5 pagutierrez
4 chage -E 2017-10-30 pagutierrez
```

1. Podemos ver con la opción -M, el número de días que puede estar con la misma contraseña.
2. Con la opción -W, establece un aviso de que la contraseña expira un número de días antes de que ocurra.
3. Con la opción -I, número de días para cambiar la contraseña una vez haya expirado la anterior, la cuenta se deshabilitará si la contraseña no ha sido cambiada.
4. Con la opción -E, fecha en la que la cuenta expira y se deshabilita de forma automática.

-. Ficheros de inicialización: son scripts shell que realizan tareas como dar valor a variables, nombrar alias, realizar funciones específicas....

Se ejecuta al hacer un <i>login</i>	<code>.bash_profile</code> en bash
en el sistema por SSH o por terminal	<code>.profile</code> en bash y sh
real	<code>.login</code> en csh
Cada vez que se ejecuta una shell,	<code>.bashrc</code> en bash
aunque no conlleve <i>login</i>	<code>.cshrc</code> en csh
Al salir del sistema el usuario	<code>.bash_logout</code> en bash
(al finalizar la sesión)	<code>.logout</code> en C csh



- **Intérprete de órdenes:** este se establece en el último campo del fichero `/etc/passwd/`. Si un usuario no tiene asignado ningún intérprete de órdenes, se usará el shell por defecto `/bin/sh`. Para privar a un usuario a la entrada al sistema se le asigna `/bin/false` o `/bin/nologin/`. También se puede establecer como shell un fichero ejecutable, pero al finalizar el sistema el usuario no llega a hacer su login.

- **Cuentas restrictivas:** permiten limitar las acciones de los usuarios del sistema. Se pueden crear de dos formas:

1. Asignar como shell un fichero ejecutable que realice una tarea determinada, y al terminar sale del sistema. Las tareas que pueden realizar quedan determinadas por nivel de identificador del usuario.
2. Usando el shell restrictivo `/bin/rbash/`, para crear dichas limitaciones, hay que copiar los ficheros que puedan ejecutar en un directorio y que su *PAHT* sea sólo ese directorio.

- Añadir un usuario al sistema, pasos a seguir:

1. Decidir el nombre de usuario, el UID, y los grupos a los que va a pertenecer (grupo **primario** y grupos **secundarios**).
2. Introducir los datos en los ficheros `/etc/passwd` y `/etc/group` (poniendo como contraseña "`*`").
3. Asignar un *password* a la nueva cuenta.
4. Si las *shadow* están activas, escribir la contraseña.
5. Establecer los parámetros de envejecimiento de la cuenta.
6. Crear el directorio `$HOME` del nuevo usuario, establecer el propietario y grupo correspondiente y los permisos adecuados.
7. Copiar ficheros necesarios por defecto (`.bash_profile`, `.bashrc`...) desde `/etc/skel/`.
8. Establecer otras facilidades: *quotas*, *mail*, permisos, etc.
9. Ejecutar cualquier tarea de inicialización propia del sistema.
10. Probar la nueva cuenta.

- **Herramientas para crear/modificar las cuentas de usuario:**

1. `adduser` o `useradd`: crear cuentas de usuario, o modificar cuentas ya existentes.
2. `usermod`: modificar cuentas.
3. `deluser` o `userdel`: eliminar cuentas.
4. `newusers`: crear cuentas de usuarios utilizando la información introducida en un fichero de texto (en *batch*), que ha de tener el formato del fichero `/etc/passwd/`.
5. `user-admin`: herramienta en modo gráfico.

- Grupos, funcionamiento de los grupos: al crear un fichero se establece como grupo propietario el grupo activo del usuario en ese momento, el grupo activo sería el grupo primario. Al determinar los permisos sobre un fichero, se usan todos los grupos de usuarios. Tipos de grupos:

1. Primarios: grupo especificado en */etc/passwd*.
2. Secundarios: otros grupos, indicados en */etc/group*.

- Comandos para grupos:

- `addgroup grupo` ⇒ crear un nuevo grupo.
- `groupmod grupo` ⇒ modificar un grupo existente.
- `delgroup grupo` ⇒ eliminar un grupo.
- `newgrp grupo` ⇒ cambiar de grupo activo (lanza un shell)
- `gpasswd grupo` ⇒ asignar una contraseña a un grupo:
 - Si el usuario no pertenece al grupo, pero el grupo tiene contraseña, se le solicita y pasa a ser su grupo activo.
- `gpasswd -a user grupo` ⇒ añadir un usuario a un grupo.
- `groups [usuario]` ⇒ grupos a los que pertenece un usuario.
- `id [usuario]` ⇒ lista el identificador del usuario y los grupos a los que pertenece.
- `grpck` ⇒ chequea la consistencia del fichero de grupos.

- Usuarios estándar:

1. root: cuenta de administrador.
2. bin (utilidades comunes de usuarios), daemon (ejecución de demonios)... son los tradicionalmente usados para poseer ficheros y ejecutar servicios.
3. mail, news, ftp: asociados con herramientas o facilidades.
4. postgres, mysql, xfs: creados por herramientas instaladas en el sistema para administrar y ejecutar sus servicios.
5. nobody o nfsbody: usado por NFS y otras utilidades, usuario sin privilegios.

Tema 5: Gestión de Recursos, para una correcta administración del sistema se implica obtener información sobre sus recursos y rendimiento.

- Definición de proceso, es un programa en ejecución.
- 1. Un sistema de tiempo compartido permite múltiples usuarios que ejecuten múltiples procesos, pero la CPU solo puede ejecutar un proceso a la vez.
- 2. La CPU conmuta rápidamente de un proceso al siguiente, ejecutando un quantum de cada proceso.
- 3. El SO es el encargado de decidir qué proceso se ejecuta en qué lugar, se logra con la planificación de la CPU.

- Tipos de procesos:

1. Procesos de usuario: son procesos creados por un usuario real.
2. Procesos demonio: no están asociados a un usuario, o asociados a uno ficticio. En su labor, son los relacionados con tareas periódicas de la administración del sistema.
3. Procesos núcleo: no asociados a un usuario. Corresponde al código Kernel y realizan tareas de administración delicadas.

- Modo de ejecución: es una distinción para proteger mejor las direcciones de memoria a las que puede acceder un proceso. Tipos:

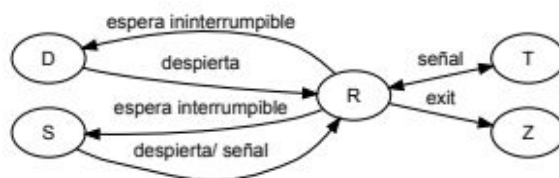
1. Modo usuario: se ejecuta código normal del programa.
2. Modo núcleo: se ejecutan las funciones del núcleo, como son las llamadas al sistema (procesos de usuario que solicitan servicios y con ello se crea por ejemplo un hilo), excepciones (situaciones excepcionales que causan excepciones hardware que requieren intervención del Kernel) e Interrupciones (los dispositivos periféricos interrumpen para notificar al Kernel de diversos sucesos).

- Monitorizar los procesos: ps, es la información sobre los procesos en ejecución.

1. USER: usuario que lanzó el programa.
2. PID: identificador del proceso.
3. PPID: identificador del proceso padre.
4. %CPU: porcentaje de la CPU consumido por este proceso.
5. %MEM: fracción de memoria consumida.
6. VSZ: tamaño virtual en KB.
7. RSS: memoria real usada en KB.
8. TTY: terminal asociado con el proceso.
9. STAT: estado del proceso.

- Estados de los procesos:

1. R: en ejecución, está listo para ejecutarse un proceso en cuanto la CPU esté libre.
2. D: Durmiendo ininterrumpible (E/S), el proceso no maneja señales, solo despertará cuando pase el evento.
3. S: Durmiendo, esperando que ocurra un evento específico.
4. T: parado (señal o trace), proceso detenido temporalmente mediante señales o porque está siendo examinado (trace). Solo volverán a ejecutarse tras otra señal.
5. Z: proceso zombie, el proceso termina correctamente pero el padre no recoge su código de error, para ver el origen del problema hay que consultar el PID del padre.
6. s: Líder de sesión, procesos que se pueden agrupar y se puede interactuar con ellos por medio de la terminal.
7. l: hijos creados con CLONE_THREAD.
8. L: un proceso pide no modificar algunas páginas de memoria mientras se hacen ciertas operaciones.
9. +: foreground, proceso de primer plano, iniciando sin &.





- Control/ gestión de la actividad de la CPU:

- Número nice y prioridad de procesos, es una planificación por prioridades dinámicas. Al lanzar un proceso se le asigna un número nice o prioridad estática, la prioridad por defecto se obtiene mediante el número nice.

1. Valores bajos (negativos), más prioridad.
2. Valores altos (positivos), menos prioridad.

Ejemplo: nice -5 nautilus: lanzar nautilus con nº nice incrementado en 5.

- Envío de señales a los procesos:

1. kill pid: mandar señal por defecto al proceso pid.
2. SIGKILL, fuerza la salida del proceso.
3. SIGSTOP, para un proceso.
4. SIGCONT, reiniciarlo.
5. Los procesos detenidos con TSTP o con STOP, se pueden reanudar con la señal CONT.
6. killall comando, permite mandar una señal a todos los procesos con un determinado nombre de comando.
7. pkill o skill, enviar una señal usando el nombre u otros atributos o criterios.
8. Los procesos en estados D o Z no se detiene pese a recibir la señal de KILL.

- Comandos para el control y gestión de procesos:

1. uptime: hora actual, cuánto tiempo lleva en marcha el sistema, número de usuarios conectados, y carga media del sistema.

```
pagutierrez@TOSHIBA:~$ uptime
13:31:05 up 1:32, 3 users, load average: 0.18, 0.19, 0.19
```

2. pstree: visualiza un árbol de los procesos de ejecución.

```
1 init--NetworkManager---2*[{NetworkManager}]
2 |--acpid
3 |--atd
4 |--avahi-daemon---avahi-daemon
5 |--bamfdemon
6 |--bluetoothd
7 |--bonobo-activati---2*[{bonobo-activat}]
8 |--console-kit-dae---64*[{console-kit-da}]
9 |--cron
10 |--cupsd
11 |--2*[dbus-daemon]
12 |--dbus-launch
13 |--dconf-service---{dconf-service}
14 |--evince---3*[{evince}]
15 |--evince---{evince}
16 |--firefox---plugin-containe---7*[{plugin-contain}]
17 | |--21*[{firefox}]
18 |--gconfd-2
```

3. top: proporciona una visión continua de la actividad del procesador, en tiempo real, mostrando las tareas que hacen más uso de la CPU. Además, permite manipular procesos de forma interactiva. Primeras cinco líneas:

```
1 top - 11:33:17 up 2:11, 4 users, load average: 0.12, 0.19, 0.35
2 Tasks: 183 total, 1 running, 181 sleeping, 0 stopped, 1 zombie
3 Cpu(s): 6.9%us, 2.6%sy, 0.0%ni, 89.8%id, 0.8%wa, 0.0%hi, 0.0%si, 0.0%st
4 Mem: 1012004k total, 970028k used, 41976k free, 8444k buffers
5 Swap: 2080760k total, 507884k used, 1572876k free, 278284k cached
6
7 PID USER PR NI VIRT RES SHR S %CPU %MEM TIME+ COMMAND
8 1001 root 20 0 170m 17m 4496 S 12 1.7 30:23.62 Xorg
9 2489 pagutier 20 0 335m 11m 6200 S 9 1.1 0:50.35 gnome-terminal
10 1545 pagutier 20 0 304m 7240 4112 S 6 0.7 4:09.81 compiz
11 2208 pagutier 20 0 392m 10m 5784 S 4 1.0 4:48.32 plugin-containe
12 2148 pagutier 20 0 881m 208m 15m S 3 21.1 7:50.88 firefox
13 5549 pagutier 20 0 763m 22m 12m S 2 2.3 0:21.16 knotify4
14 5677 root 20 0 0 0 0 S 0 0.0 0:00.06 kworker/2:1
15 5693 pagutier 20 0 19460 1500 1060 R 0 0.1 0:00.12 top
16 1565 pagutier 20 0 320m 6080 4636 S 0 0.6 0:05.66 gnome-power-man
17 1611 pagutier 20 0 534m 10m 4932 S 0 1.0 0:38.68 dropbox
18 1917 pagutier 20 0 360m 6856 3688 S 0 0.7 0:28.67 ubuntuone-synco
19 4765 pagutier 20 0 585m 40m 8980 S 0 4.1 0:52.57 evince
20 197 root 20 0 0 0 0 S 0 0.0 0:02.40 usb-storage
21 457 messageb 20 0 24892 1684 668 S 0 0.2 0:09.26 dbus-daemon
22 507 avahi 20 0 32404 1152 800 S 0 0.1 0:09.01 avahi-daemon
23 513 root 20 0 162m 2900 2292 S 0 0.3 0:04.79 NetworkManager
24 1086 root 20 0 15784 448 364 S 0 0.0 0:03.76 irqbalance
25 1478 pagutier 20 0 238m 3908 3144 S 0 0.4 0:01.42 gnome-session
26 1515 pagutier 20 0 26708 2032 560 S 0 0.2 0:12.53 dbus-daemon
27 1531 pagutier 20 0 464m 5392 4264 S 0 0.5 0:20.41 gnome-settings-
28 1556 pagutier 20 0 671m 12m 6984 S 0 1.3 4:44.00 nautilus
29 1558 pagutier 20 0 398m 9304 5460 S 0 0.9 0:30.69 gnome-panel
30 1559 pagutier 20 0 398m 8876 5552 S 0 0.9 0:09.11 nm-applet
```

- a) Estadísticas de uptime.
- b) Resumen del número de procesos en el sistema.
- c) Estado actual de la CPU: porcentaje en uso modo usuario (us), modo sistema o núcleo (sy), proceso valor nice positivo (ni), procesos ociosos (id), procesos esperando eventos E/S (wa), tratando interrupciones (hardware o software, hi o si), espera involuntaria en virtualización (st).
- d) Estado actual de la memoria física: total disponible, usada, libre, usada en buffers.
- e) Espacio swap (espacio de intercambio de un disco): total disponible, usada, libre, usada en buffers, usada en caché de página.

Los datos de la parte inferior son similares a los de pc, excepto SHR, es la memoria compartida disponible para ser utilizada. Características:

- a) Procesos ordenados decrecientemente por uso de CPU.
- b) Lista actualizada interactivamente, normalmente cada 5s.

4. vmstat: información sobre la memoria virtual (también procesos), desde el último reinicio:

- a) r: número de procesos esperando su tiempo de ejecución.
- b) b: número de procesos en espera ininterrumpible.
- c) us: tiempo de usuarios como porcentaje en tiempo total (modo usuario).
- d) sy: tiempo de sistema como porcentaje de tiempo total (modo núcleo).
- e) id: tiempo de inactividad como porcentaje de tiempo total.
- f) wa: tiempo usado en espera de E/S.

```
1 pagutierrez@TOSHIBA:~$ vmstat 2 5 # 5 informes cada 2 segundos
2 procs-----memory-----swap-----io-----system-----cpu-----
3 r b swpd free buff cache si so bi bo in cs us sy id wa
4 0 0 465400 60300 15376 196640 11 25 140 58 213 319 10 3 83 4
5 0 0 465400 60236 15384 196628 0 0 0 10 865 658 3 2 95 1
6 0 0 465400 60096 15384 196716 0 0 0 50 858 704 4 1 95 0
7 0 0 465400 60128 15384 196672 0 0 0 0 787 657 3 2 95 0
8 0 0 465400 59384 15392 196744 0 0 0 38 787 907 13 2 85 0
```

5. at: ejecuta tareas a una determinada hora, puede recibir un fichero de texto con las órdenes a ejecutar. Órdenes:

- a) atd: demonio que ejecuta las órdenes.
- b) atq: consulta la lista de órdenes.
- c) atrm: eliminar órdenes.

```
1 pagutierrez@pagutierrez--TOSHIBA:~$ at 14:38
2 warning: commands will be executed using /bin/sh
3 at> echo "HOLA" > /tmp/p2
4 at> <EOT>
5 job 10 at Sat Mar 8 14:38:00 2016
6 pagutierrez@pagutierrez--TOSHIBA:~$ date
7 sáb mar 8 14:37:47 CET 2016
8 pagutierrez@pagutierrez--TOSHIBA:~$ cat /tmp/p2
9 cat: /tmp/p2: No existe el fichero o el directorio
10 pagutierrez@pagutierrez--TOSHIBA:~$ date
11 sáb mar 8 14:38:01 CET 2016
12 pagutierrez@pagutierrez--TOSHIBA:~$ cat /tmp/p2
13 HOLA
```


6. cron: se usa para ejecutar tareas periódicamente. Órdenes:

- a) crond: demonio encargado de ejecutar las órdenes.
- b) crontab: establecer tareas a ejecutar, su formato es:

crontab: minuto hora día-mes mes día-semana [user] comando

Es interesante saber que los domingos son el día 0 y 7 de la semana. Si la máquina no está encendida cuando se ha de lanzar el proceso, cron no lo lanza.

```
1 # Hacer una copia de seguridad del home cada semana
2 0 5 * * 1 tar -zcf /var/backups/home.tgz /home/
3 # Otras tareas
4 5 9 * * * $HOME/tareadiaria # 9:05
5 15 14 1 * * $HOME/tareames # 14:15 el día 1 cada mes
6 0 22 * * 1-5 $HOME/tareasemanal # 22:0 de lunes a viernes
7 21 0-23/2 * * $HOME/tareacada2horas # 0h, 2h, 4h, 6h, y 21m
8 5 4 * * sun $HOME/tareadomingos # Domingos a las 4:05
9 0 9 1 * 5 $HOME/otratarea # A las 9:00h el día 1
0 # de cada mes 0 los viernes
```

c) anacron: no asume que la máquina está siempre encendida. Combina el uso de scripts al inicio con el uso de cron.

- Rastreo de señales y llamadas al sistema: se da uso al comando strace, nos permite observar qué es lo que está haciendo un proceso. Muestra cada llamada al sistema que hace y cada señal que recibe.

```
1 while 1
2     mkdir adir
3     cd adir
4     touch afile
5 end
```



- Memoria: intercambio y paginación con la memoria virtual para alojar procesos. Debemos gestionar por tanto la RAM y la zona de intercambio. Comando usado es vmstat: su contenido es el siguiente:

1. swpd: cantidad de memoria virtual (intercambio) ocupada.
2. free: cantidad de memoria virtual sin usar.
3. buff: cantidad de memoria empleada como buffers para E/S.
4. cache: la cantidad de memoria empleada como caché de disco.
5. si: cantidad de memoria traída del espacio de intercambio desde disco en KB/s.
6. so: cantidad de memoria intercambiada al disco en KB/s.
7. bi: bloques recibidos desde un dispositivo de bloques (en bloques/s).
8. bo: bloques enviados a un dispositivo de bloques (en bloques/s).
9. in: número de interrupciones por segundo, contando el reloj.
10. cs: número de cambios de contexto por segundo.

```
1 pagutierrez@TOSHIBA:~$ vmstat 2 2
2
3 r b swpd free buff cache si so bi bo in cs us sy id wa
4 0 0 465400 60300 15376 196640 11 25 140 58 213 319 10 3 83 4
5 0 0 465400 60236 15384 196628 0 0 0 10 865 658 3 2 95 1
```

- Espacio para paginación, el tamaño adecuado para la paginación depende de la memoria requerida por los procesos, número de procesos simultáneos, etc. y por consiguiente depende de la demanda del sistema. Por ello, puede tener una partición de intercambio o un fichero de intercambio. Se puede controlar con número de prioridad en /etc/fstab. Comandos a usar:

1. swapon -s: da un listado de las particiones o ficheros activos.
2. swapon /dev/sdd1: activar una determinada partición, en este caso sdd1.
3. swapoff /dev/sdd1: desactivar una determinada partición. Para crear un fichero de paginación.

```
1 sudo dd if=/dev/zero of=./fichero_swap bs=1048576 count=1024
2 sudo mkswap ./fichero_swap
3 sudo sync
4 sudo swapon ./fichero_swap
```

4. free: obtener información sobre el uso de memoria (mismos campos que top).

```
1 pedroa@pedroa-laptop ~$ free
2
3 total used free shared buffers cached
4 Mem: 6014848 5572868 441980 0 651400 2002768
5 -/+ buffers/cache: 2918700 3096148
6 Swap: 4112636 0 4112636
```

5. df: muestra la capacidad, el espacio libre y el punto de montaje de cada sistema de ficheros del equipo.

```
1 pagutierrez@TOSHIBA:~$ df -h
2 S.ficheros      Tam.  Usado Disp.  % Uso Montado en
3 /dev/sda4       114G   28G   81G   26% /
4 none           487M   720K  486M   1% /dev
5 none           495M   384K  494M   1% /dev/shm
6 none           495M   100K  495M   1% /var/run
7 none           495M     0   495M   0% /var/lock
8 /dev/sdd6       367G   74G   275G   22% /media/cade...
9 /dev/sdd5       91G    12G   75G   14% /media/4396...
```

- Si el sistema de ficheros raíz se quedase sin espacio el sistema tendría problemas. Pej., no podría arrancar, (¿por qué?).
- “-i” nos permite mostrar información sobre los nodos-i.

```
1 pagutierrez@TOSHIBA:~$ df -i /dev/sda4
2 S.ficheros      Inodos  IUsado  ILibre IUsd% Montado en
3 /dev/sda4       7569408 295620 7273788 4% /
```

6. du: muestra el espacio usado por cada subdirectorio del directorio actual. Al no poner la opción --max-depth=1 nos muestra todas las carpetas. La última línea es el acumulado.

```
1 pagutierrez@TOSHIBA:~/PAS$ du -h --max-depth=1
2 4,8M      ./Guiones
3 268K      ./Programa1112
4 204K      ./GuiaDocente
5 39M      ./MaterialAdicional
6 46M      .
```

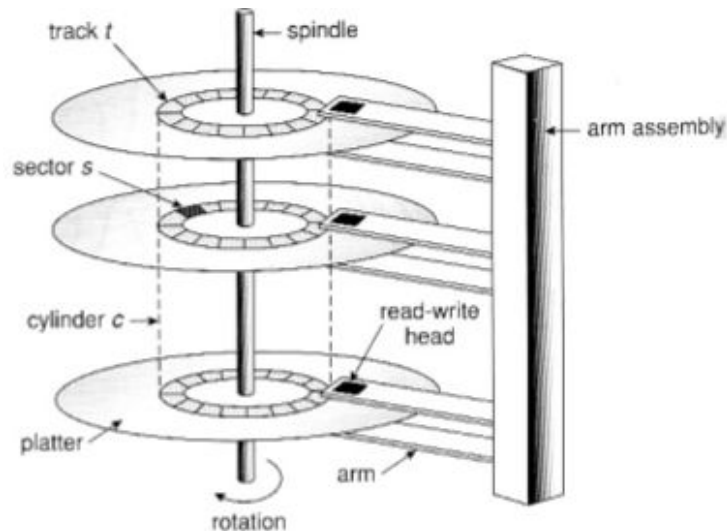
7. iostat intervalo numero: presenta estadísticas sobre la CPU, los dispositivos y particiones de E/S. Apartados de el comando:

- a) tps: número de transferencias por segundo.
- b) Blk_read: número de bloques leídos por segundo.
- c) Blk_wrtn/s: número de bloques escritos por segundo.
- d) Blk_read: número total de bloques leídos.
- e) Blk_wrtn: número de bloques escritos.

```
1 pedroa@pedroa-laptop ~ $ iostat
2 Linux 3.11-2-amd64 (pedroa-laptop) 14/03/15      _x86_64_      (8 CPU)
3
4 avg-cpu:  %user   %nice %system %iowait  %steal   %idle
5            2,48    0,00   0,45   0,08    0,00   96,99
6
7 Device:            tps    kB_read/s    kB_wrtn/s    kB_read    kB_wrtn
8 sda                23,40        307,28        319,16    2279946    2368084
9 sdb                 0,71         4,01         0,62     29747     4588
```

Tema 6: Organización de sistemas de ficheros y discos.

- Introducción: La función principal de un disco duro es almacenar la información del PC cuando no se encuentra conectado a corriente eléctrica. También podría servir como extensión a la memoria RAM, gracias al mecanismo de memoria virtual.
- Discos rígidos: funcionan de forma parecida a un tocadiscos, mientras que los discos SSD utilizan una memoria formada por semiconductores para almacenar la información.



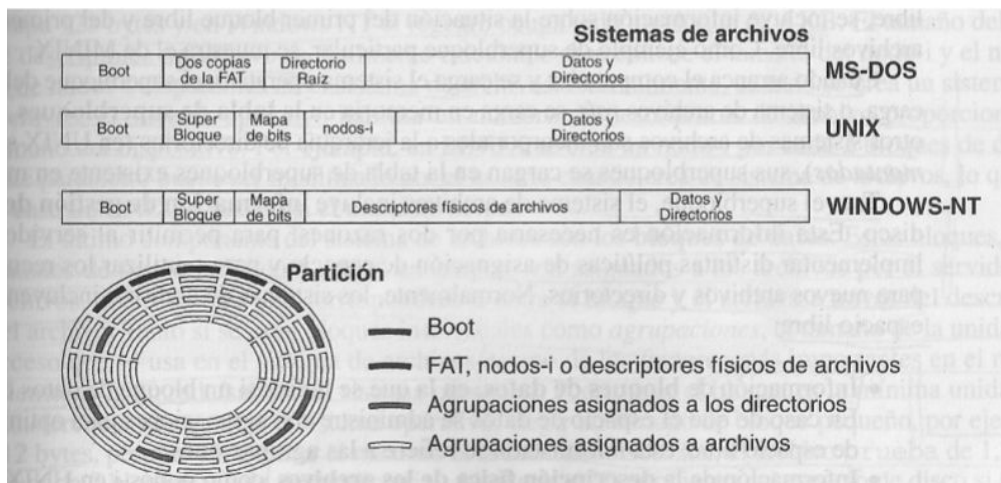
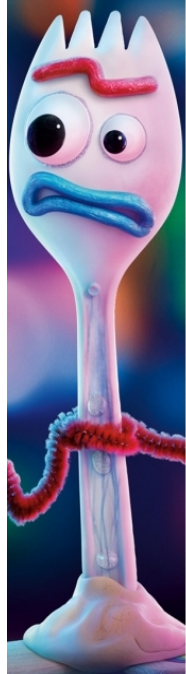
- Composición de un disco duro:

1. Plato: Cada uno de los discos que se encuentran apilados en su interior, cubiertos de un material magnetizable. La escritura cambia de estado este material.
2. Cabezal: es un brazo que se mueve sobre el plato, permite acceder a cualquier punto de los mismos.
3. Pista: se trata de cada una de las líneas esféricas que se pueden formar sobre cada plato.
4. Cilindro: Conjunto de varias pistas que se encuentran una encima de otra.
5. Sector: cada una de las divisiones que se hace de la circunferencia que se forma en el disco

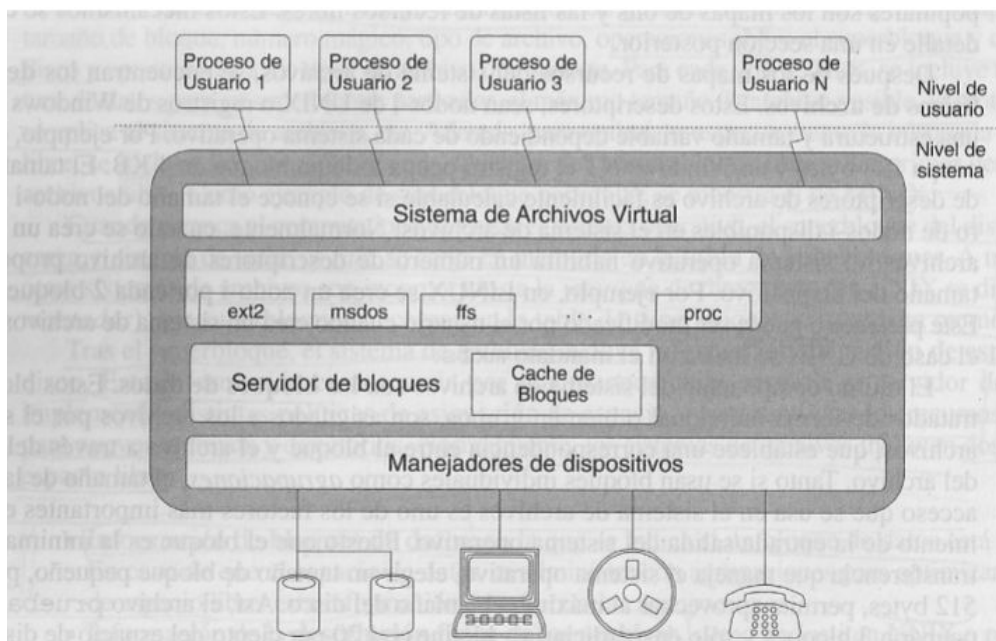
- Archivos: es una unidad de almacenamiento lógico no volátil que agrupa un conjunto de información relacionada entre sí bajo un mismo nombre. Para acceder a un archivo puede ser secuencial (para acceder a una posición hay que conocer la anterior) o directo/aleatorio (se puede acceder a cualquier posición).

- Sistema de Archivos:

1. Partición: porción de un disco a la que se le dota de una identidad propia y que se manipula como una entidad lógica independiente.
2. Bloque: agrupación lógica de sectores físicos del disco, la cual supone la unidad de transferencia mínima que usa el SA. El tamaño de bloque es un parámetro decisivo que afecta a la eficiencia del acceso a disco y a la fragmentación del mismo.
3. Agrupación: conjunto de bloques gestionado como una unidad lógica de almacenamiento.
4. El bloque de carga (boot o Volume Boot Record) contiene código ejecutado al arrancar el ordenador por el iniciador ROM utilizando esa partición. Se suele incluir en las particiones para así mantener una estructura uniforme. Se le añade un número mágico, el cual será comprobado por el iniciador ROM para demostrar que el bloque de carga es válido.
5. Metainformación: describe el SA (Sistema de Archivos) y la distribución de sus componentes. Es necesario para poder acceder a los datos.
6. Superbloque: se mantiene una serie de información común para todos los SAs u una entrada característica para cada tipo de SA. Al arrancar la máquina, los superbloques de todos los SAs que son cargados se mantienen en memoria.
7. Descriptores físicos de archivos: describen cada uno de los archivos almacenados, tienen una estructura y tamaño muy ligada al SO. El número de descriptores debe ser proporcional al tamaño total del disco.
8. Gestión del espacio libre: distintos mecanismos permiten gestionar el espacio libre. Por ende se gestionan dos tipos de recursos:
 - a) Mapas de bloques: indican qué bloques o agrupaciones están libres.
 - b) Mapas de descriptores de archivos: indican qué descriptores de archivos están libres.
9. Bloques de datos: es dónde se almacena realmente información.



- Servidor de archivos: es el componente del SO que se encargará de gestionar el acceso a archivos. Se sigue una organización por capas, proporcionan servicios a los niveles superiores los niveles inferiores, y en cada nivel se aumenta la abstracción de las operaciones.



1. Sistema de archivos virtual: proporciona la interfaz para las llamadas de E/S que deseen realizar los procesos de usuario, interactuando con el módulo de organización de archivos. Cumple funciones de manejo de directorios, gestión de nombres... para ello es necesario usar una estructura adicional, que incluye las características comunes a todos los sistemas de archivos y un enlace al descriptor de archivo particular. Los nodos virtuales contienen la siguiente información:

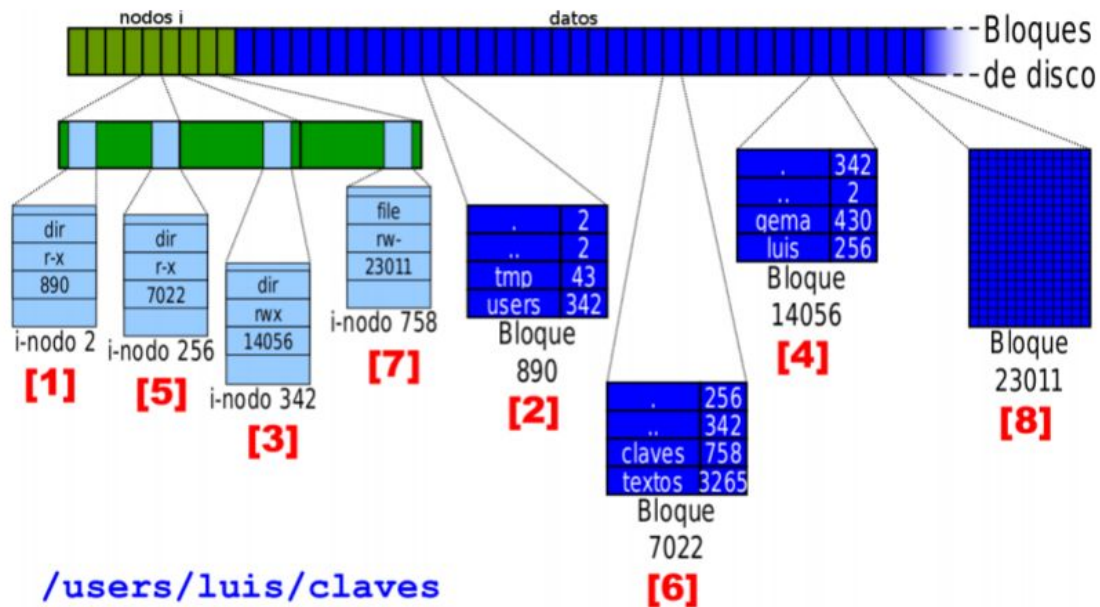
- a) Atributos del archivo.
- b) Puntero al nodo-i real.
- c) Punteros a funciones que realizan las operaciones genéricas de cualquier SA.
- d) Punteros a funciones que realizan las operaciones propias del SA concreto.

2. Módulo de organización de archivos, se implementa por separado para cada tipo de SA. Relaciona la imagen lógica de un archivo con su imagen física, traduciendo direcciones lógicas del archivo a las direcciones físicas del dispositivo. Se basa en la información de los nodos-i y utiliza los servicios del servidor de bloques para realizar las operaciones correspondientes.

3. Servidor de bloques: en este nivel se emiten los mandatos genéricos para leer y escribir bloques en los manejadores de dispositivos. Se traducirán en llamadas al manejador específico del Sistema de Archivos. En este nivel se realiza la caché de bloques, es decir se almacenan datos para que las solicitudes futuras de esos datos se pueden atender con mayor rapidez.

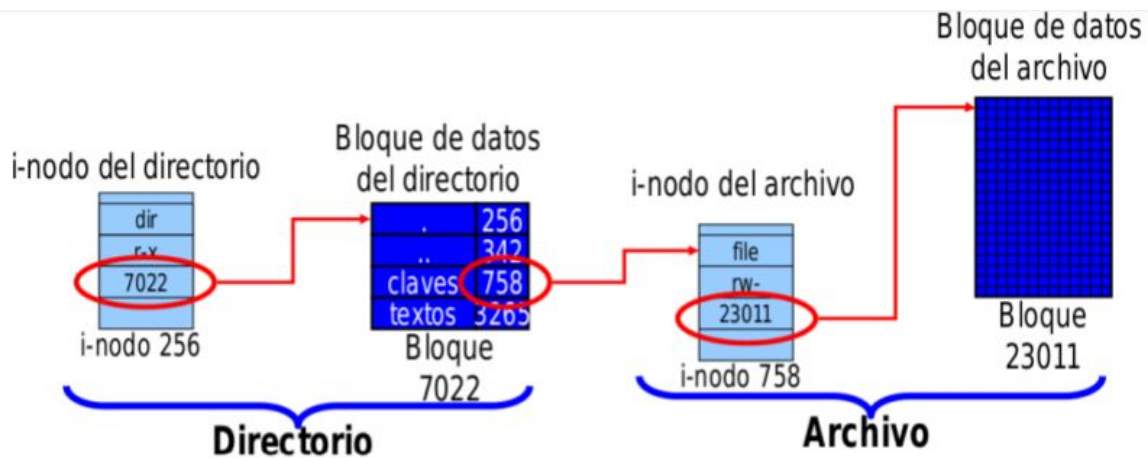
4. Manejador de dispositivos: traducen órdenes de E/S de alto nivel a un formato que puede entender el dispositivo (dependiente del hardware).

- Directorios: es un fichero con un formato determinado, en su contenido guarda una serie de entradas (registros), una por cada fichero contenido en él. Cada registro contiene mínimo el nombre del fichero y el puntero al descriptor físico correspondiente. Ejemplo de uso, con la siguiente ruta `/users/luis/claves` se interpreta de forma recursiva:

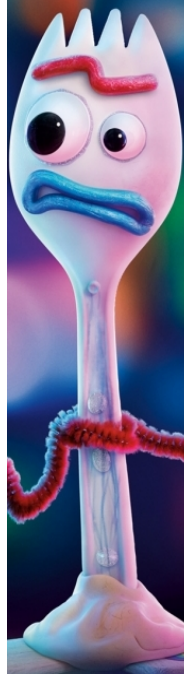


1. Traer a memoria bloque del i-nodo 2.
2. Se busca dentro `users` y se obtiene el i-nodo 342.
3. Traer a memoria bloque del i-nodo 342.
4. Se busca dentro `luis` y se obtiene el i-nodo 256.
5. Traer a memoria bloque del i-nodo 256.
6. Se busca dentro `claves` y se obtiene el i-nodo 758.
7. Al leer el i-nodo 758, se detecta que es un fichero y ya se tienen dónde están los datos del archivo.
8. Leer los bloques del fichero.

Importante, la llamada open() termina con la lectura del i-nodo, la verificación de permisos se hace con los datos del i-nodo y un directorio no es un i-nodo.

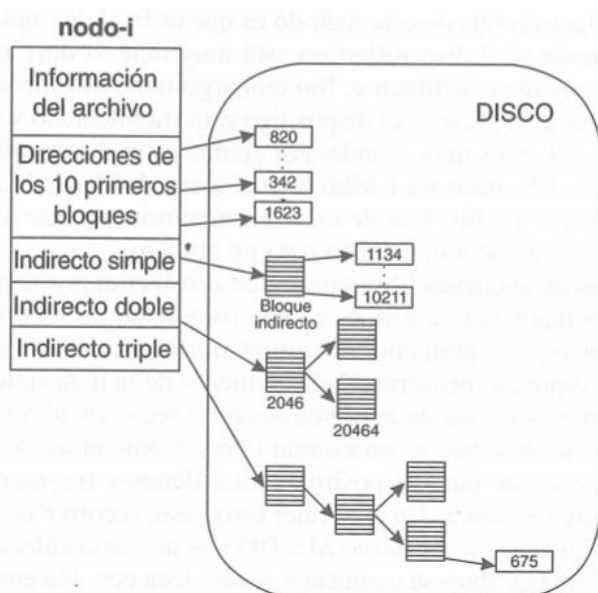


- Asignación y mecanismos de asignación: la asignación es como se hace la correspondencia entre los bloques físicos del disco y los bloques lógicos del archivo. Mecanismos de asignación existen dos:
 1. Asignación de bloques contiguos: actualmente está en desuso.
 2. Asignación de bloques no contiguos: los bloques del archivo se encuentran en cualquier posición del disco. Para tener constancia de qué bloques no contiguos pertenecen a cada archivo, se utilizan listas enlazadas o índices.
 - a) Lista enlazada: cada bloque tiene un apuntador al siguiente bloque que seguiría en el archivo. El descriptor del archivo solo debe incluir la referencia al primer bloque.
 - i) Tabla de asignación de archivos: es una variación del método lista enlazada, los apuntadores se almacenan en una tabla independiente de los bloques. La tabla se aloja en el caché para mejorar las prestaciones y se mantiene una copia doble en el disco para mayor fiabilidad.
 - b) Índices: Los punteros a los bloques están juntos y contiguos en una localización concreta, los bloques índice, y cada archivo tiene un bloque índice.
 - i) Índice multinivel: consiste en introducir n niveles de apuntadores, de manera que los apuntadores del descriptor apuntan a otros.



c) La solución impuesta por Linux es un esquema híbrido, bloques de datos o bloques índice y, en ext4 y NTFS existen los extents (bloques índice especiales que marcan una zona contigua al disco “numeroBloqueInicial, numeroBloques”), por cada nodo-i hay que incluir:

- i) Punteros directos a los 10 primeros bloques.
- ii) Puntero a un bloque índice de primer nivel.
- iii) Puntero a bloque índice de segundo nivel.
- iv) etc.



- Gestión del espacio libre: se necesita para asignar espacio a los archivos nuevos o a los que se les desea añadir datos. Se mantienen mapas de recursos, implementados como mapas de bits o listas de recursos libres, con esto se incluye un bit por recurso este será 1 si el recurso está libre, y un 0 en caso contrario.

- Lista de recursos libres, se mantiene una lista de apuntadores a los recursos libres, acto seguido si se ocupa el recurso es borrado de la lista. Como solución a los discos que tienen mucho espacio libre, y tienen que cargar la lista, se incluye un número de bloques consecutivos en la lista.

- Incremento de las prestaciones: el acceso a memoria se realiza en el orden de nanosegundos, y el acceso en disco en el orden de milisegundos. Como almacenamiento intermedio de los datos se usa la caché cargada en Memoria Principal, se almacena toda la caché en bloques:
 1. Si el bloque está en MP, las operaciones se realizarán en MP. Posteriormente se moverán los bloques de MP al dispositivo. Si la caché está llena, hay que eliminar algún bloque, suelen eliminarse los bloques que llevan mucho tiempo sin usarse. Diferentes bloques:
 - a) Bloques sucios, cambiados en caché pero no en el disco. Posee distintas políticas a la hora de mantener la coherencia:
 - i) Escritura inmediata: siempre actualizado.
 - ii) Escritura diferida: actualizamos cuando el bloque salga de la caché.
 - iii) Escritura periódica: establecer un tiempo periódico para las actualizaciones, tiene un compromiso entre rendimiento y fiabilidad. Conlleva una reducción de los posibles daños por caídas.
 - b) Se pueden distinguir entre bloques especiales (directorios, nodos-i o bloques índice) y bloques de datos. Tienen una escritura inmediata.
 - c) No se debe quitar un disco del sistema sin antes volcar los datos de la caché (se usa el comando sync).