

# Tema 7. Sistemas de entrada/salida

## Índice

- Introducción
- Estructura y componentes del sistema de entrada/salida
- Dispositivos de almacenamiento secundario
- Dispositivos de almacenamiento terciario
- El reloj
- El sistema de E/S en Linux
- El sistema de E/S en Windows NT/2K
- Servicios de E/S en POSIX

## Índice

- Dispositivos de almacenamiento secundario
  - Discos
    - Estructura física de los discos
    - Formateo de discos
      - Formateo físico
      - Creación de particiones
      - Formateo lógico

# Tema 7. Sistemas de entrada/salida

## Resultados de aprendizaje

- Explicar mecanismos de gestión de gestión y planificación de entrada/salida

# Tema 7. Sistemas de entrada/salida


## Bibliografía

- J. Carretero et al. Sistemas Operativos: Una Visión Aplicada. McGraw-Hill. 2ª edición 2007. Capítulo 8.



# Tema 7. Sistemas de entrada/salida

## Índice

-  ■ Introducción
- Estructura y componentes del sistema de entrada/salida
- Dispositivos de almacenamiento secundario
- Dispositivos de almacenamiento terciario
- El reloj
- El sistema de E/S en Linux
- El sistema de E/S en Windows NT/2K
- Servicios de E/S en POSIX



# Introducción

## ■ El SO necesita:

### ◆ Dispositivos de almacenamiento de datos

- Almacenamiento secundario (discos)
- Almacenamiento terciario (cintas)

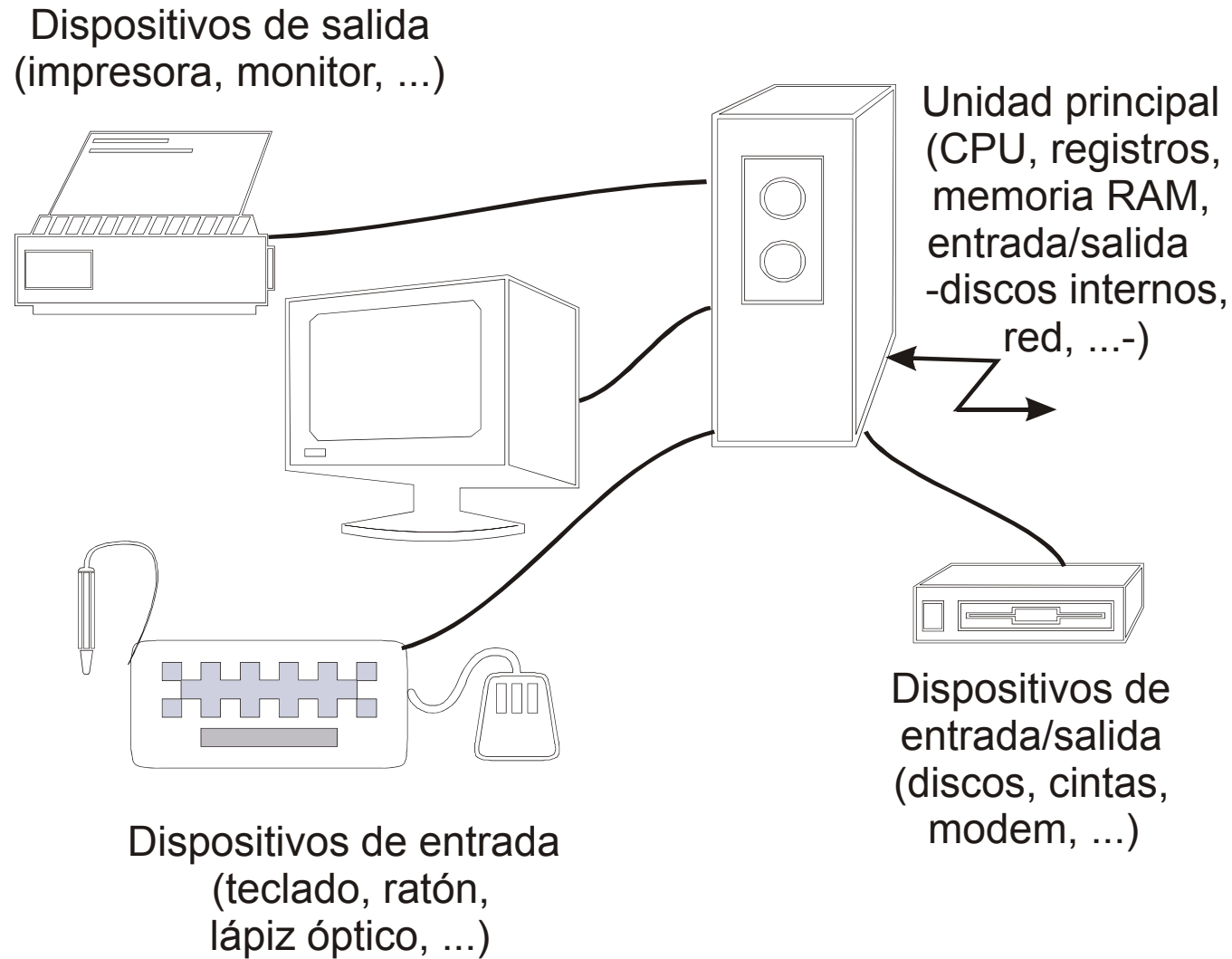
### ◆ Dispositivos de interacción con el usuario

- Periféricos de interfaz de usuario (ratones, pantallas, impresoras, escáner, lápices ópticos, etc.)

### ◆ Dispositivos de comunicaciones

- Conexión con otras máquinas (tarjetas de red, modems, etc.)

# Estructura de una computadora



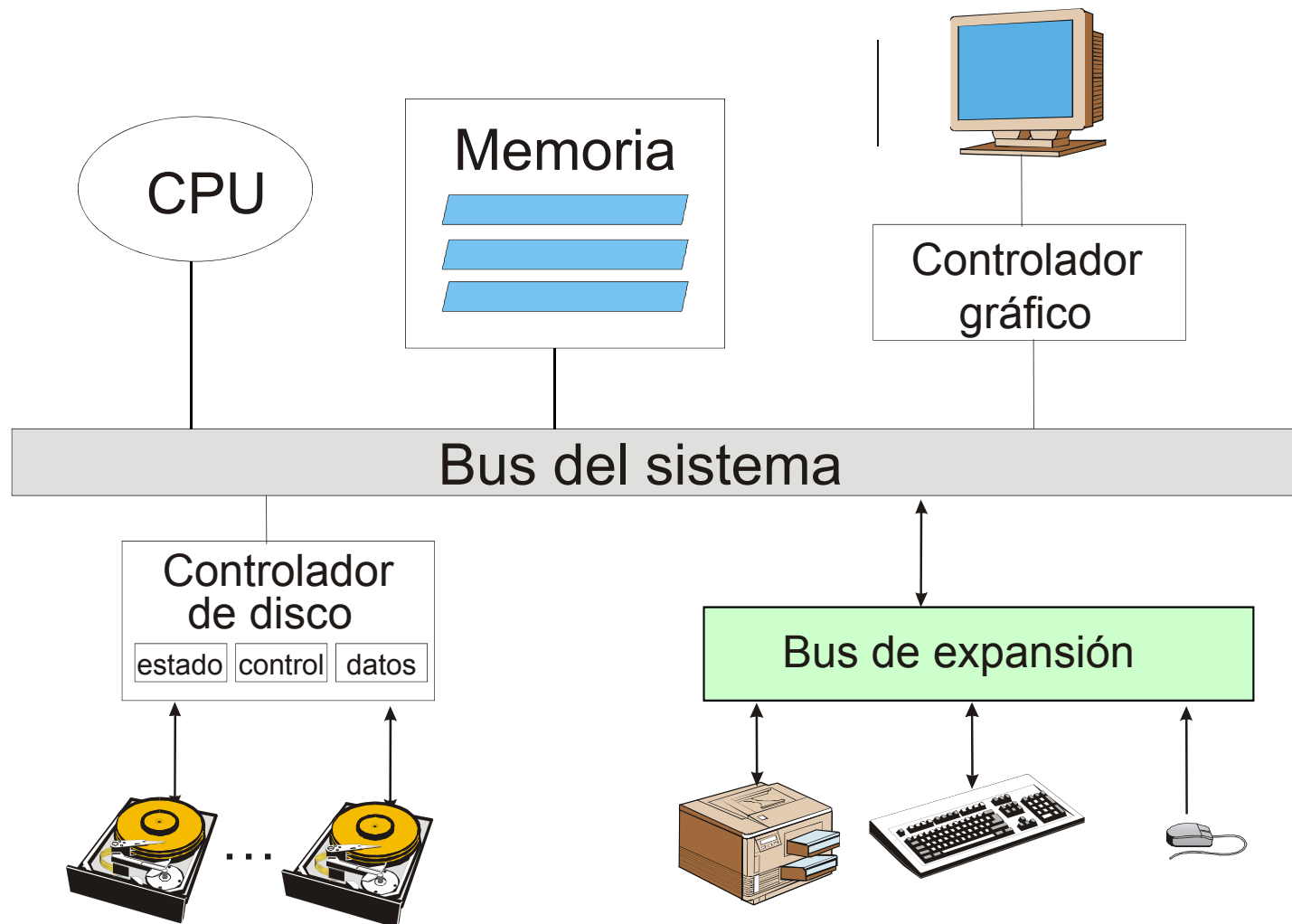


# Estructura de una computadora

- **Dispositivos por bloques y por caracteres:**
  - ◆ **Dispositivos por bloques**
    - Manejan la información en unidades de tamaño fijo (bloques)
    - Ejemplos: discos y cintas
  - ◆ **Dispositivos por caracteres**
    - Gestionan flujos de caracteres de forma lineal sin estructura tipo bloque ni tamaño fijo
    - Ejemplos: pantallas, teclados, impresoras, modems, tarjetas de red, etc.



# Conexión de dispositivos de E/S





# Conexión de dispositivos de E/S

## ■ Controladores de dispositivos:

- ◆ Un controlador puede controlar uno o más dispositivos
- ◆ Funciones:
  - Transferir información entre MC y periféricos
  - Sincronizar la velocidad del procesador con la del periférico
  - Detectar posibles errores que se produzcan en el acceso al periférico



# Funciones del sistema de E/S

- **¿Qué es el sistema de entrada/salida (E/S)?**
  - ◆ Parte del SO que ofrece:
    - Visión lógica simplificada de los dispositivos de E/S
    - Mecanismos de abstracción de los detalles físicos de éstos
- **Problema de los dispositivos de E/S:**
  - ◆ Son muy lentos
  - ◆ El cuello de botella más importante es el sistema de E/S
  - ◆ El SO necesita optimizar los mecanismos de E/S

# Funciones del sistema de E/S

## ■ Jerarquía del almacenamiento





# Funciones del sistema de E/S

## ■ Objetivos del sistema de E/S:

- Facilitar el manejo de los dispositivos de E/S
- Optimizar la E/S del sistema
- Proporcionar dispositivos virtuales para conexión de dispositivos sin tener que remodelar el SO
- Permitir la conexión de dispositivos nuevos de E/S



# Funciones del sistema de E/S

- **Mecanismos de incremento de prestaciones:**
  - Acceso directo a memoria
  - Caché en el controlador de disco
  - Caché de disco en memoria central
  - Solapamiento de búsquedas y transferencias



# Funciones del sistema de E/S

- **Mecanismos de incremento de prestaciones (cont.):**
  - ◆ Caché en el controlador de disco:
    - Memoria interna del controlador
    - Contiene copia de algunos bloques (o pistas) del disco
    - Lecturas anticipadas de datos: Principio de localidad de referencias



# Funciones del sistema de E/S

- **Mecanismos de incremento de prestaciones (cont.):**
  - ◆ **Caché de disco en memoria central (MC):**
    - Memoria intermedia situada en MC para bloques de disco
    - Memoria de tamaño variable dependiendo del espacio libre para procesos de usuario
    - Contiene copia de algunos bloques del disco
    - Lecturas anticipadas de datos: Principio de localidad de referencias
    - Estrategia de reemplazo de bloques: Generalmente LRU






# Funciones del sistema de E/S

- **Mecanismos de incremento de prestaciones (cont.):**
  - ◆ Solapamiento de búsquedas y transferencias:
    - Un controlador puede controlar varios dispositivos
    - Búsqueda de información en dispositivos lenta
    - Simultaneidad de búsquedas en varios dispositivos y de una transferencia de otro dispositivo → Optimización

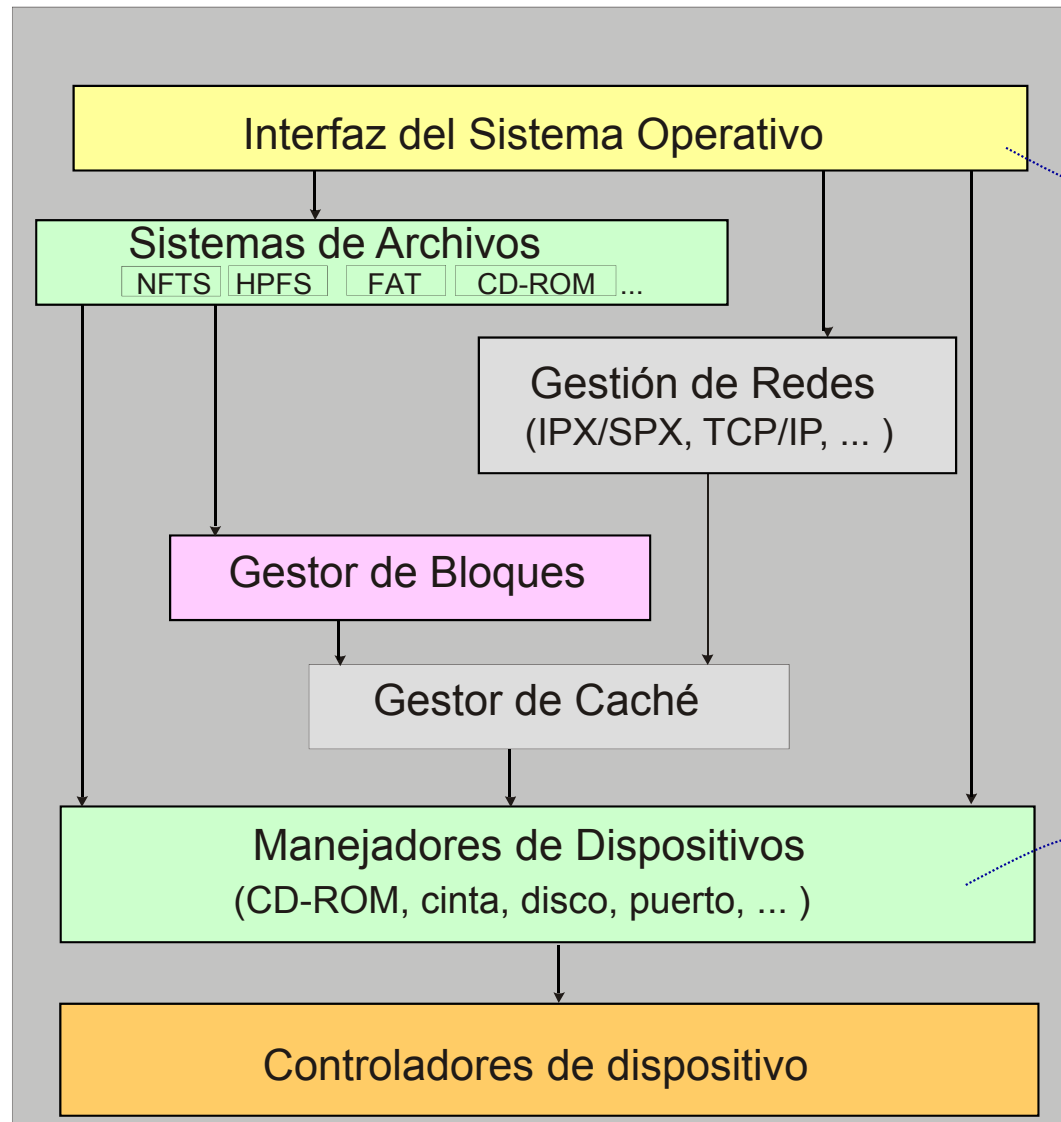


# Tema 7. Sistemas de entrada/salida

## Índice

- Introducción
-  ■ Estructura y componentes del sistema de entrada/salida
- Dispositivos de almacenamiento secundario
- Dispositivos de almacenamiento terciario
- El reloj
- El sistema de E/S en Linux
- El sistema de E/S en Windows NT/2K
- Servicios de E/S en POSIX

# Estructura del sistema de E/S



*Llamadas al sistema*

*Drivers*



# Estructura del sistema de E/S

## ■ Componentes:

### ◆ Interfaz del SO:

- Proporciona servicios de E/S a las aplicaciones y una interfaz homogénea para comunicarse con manejadores de dispositivos

### ◆ Sistemas de ficheros (SF):

- Proporciona una interfaz homogénea para acceder a los diferentes SF del SO (SV, NTFS, FAT, ...) → SF virtuales
- Traduce las operaciones del formato del usuario al de bloques

### ◆ Gestor de redes:

- Proporciona una interfaz homogénea para acceder a los diferentes sistemas de red del SO (TCP/IP, Novell, ...)



# Estructura del sistema de E/S

## ■ Componentes (cont.):

### ◆ Gestor de bloques:

- Atiende peticiones a nivel de bloque
- Interacciona con la caché de bloques para optimizar la E/S

### ◆ Gestor de caché:

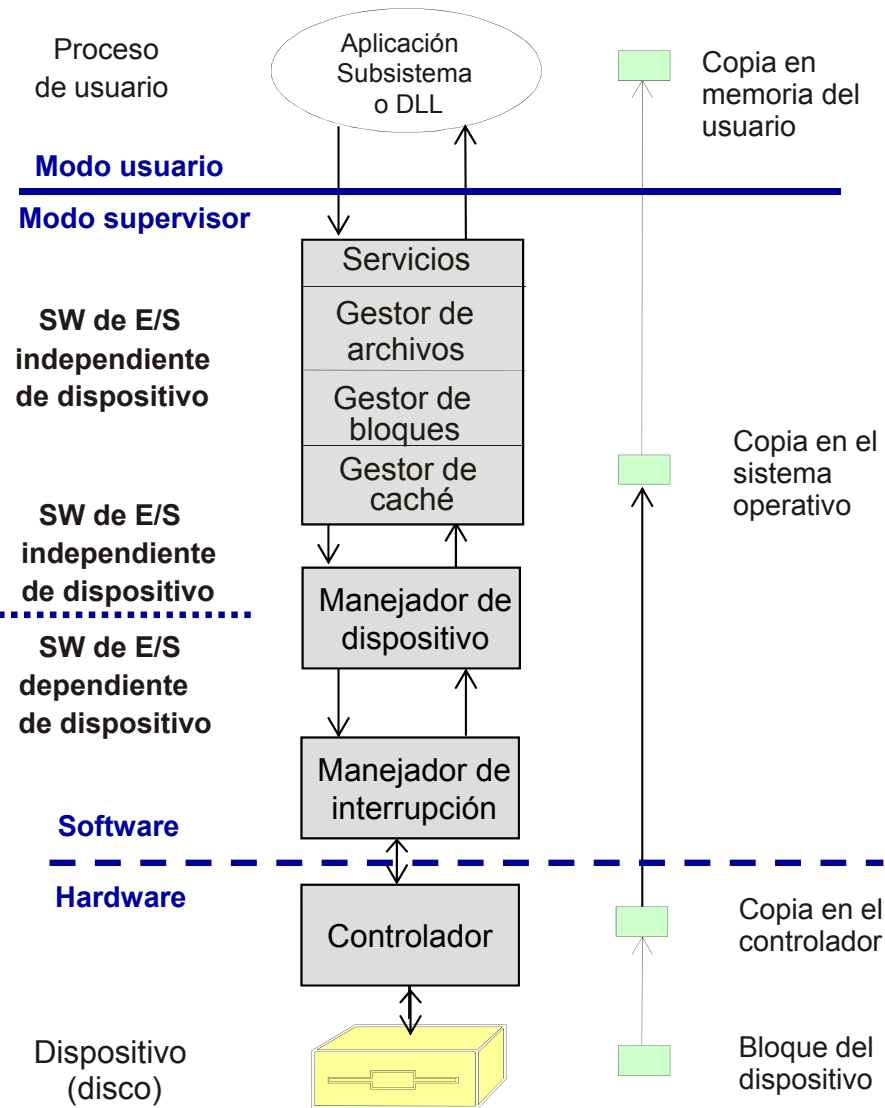
- Optimiza la E/S para dispositivos de tipo bloque mediante caché

### ◆ Manejadores de dispositivos:

- Traduce las operaciones sobre dispositivos a operaciones de control de cada dispositivo particular



# Estructura del software de E/S



# Estructura del software de E/S

## ■ Manejadores de interrupción y de dispositivos:

### ◆ Manejador de interrupción:

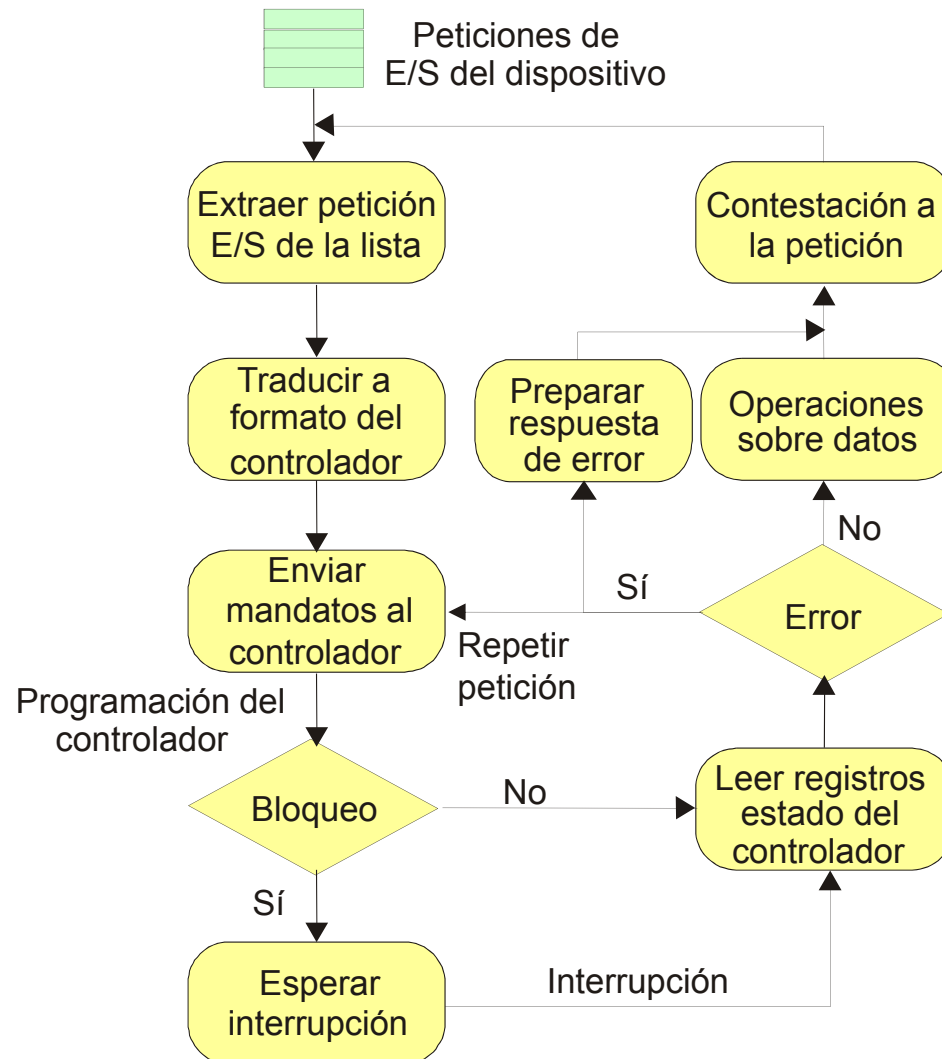
- Tratan las interrupciones de los controladores de dispositivos → Ejecución del manejador de interrupción
- Comunican el evento al manejador de dispositivo

### ◆ Manejador de dispositivo:

- Acepta peticiones del SW de E/S independiente del dispositivo
- Traduce las peticiones a formato del controlador
- Planifica el acceso de peticiones al dispositivo
- Envía las órdenes al controlador
- Espera a que se cumplan
- Comprueba el estado de la operación cuando llega la interrupción
- Gestiona los errores, si existen, y los resuelve, si es posible

# Estructura del software de E/S

- Operaciones de un manejador de dispositivo:







# Estructura del software de E/S

- **SW independiente del dispositivo:**
  - ◆ **Función:**
    - Ejecutar las funciones de E/S comunes a todos los dispositivos a través de una interfaz común
  - ◆ **Proporciona:**
    - Acceso a nivel de bloques o caracteres
      - ✓ Tamaños de acceso genéricos
    - Almacenamiento intermedio en memoria del SO
      - ✓ Optimización de la E/S evitando accesos a dispositivos
      - ✓ Oculta diferencias de velocidad
      - ✓ Facilita la implementación de la compartición



# Estructura del software de E/S

- **SW independiente del dispositivo (cont.):**
  - ◆ Proporciona (cont.):
    - Gestión de dispositivos
      - ✓ Nombrado: Traducción de nombres de usuario a identificativos del sistema
      - ✓ Protección
      - ✓ Control de acceso
    - Planificación de la E/S
    - Gestión de errores



# Tema 7. Sistemas de entrada/salida

## Índice

- Introducción
- Estructura y componentes del sistema de entrada/salida
- ☞ ■ Dispositivos de almacenamiento secundario
  - Discos
    - Estructura física de los discos
    - Formateo de discos
      - Formateo físico
      - Creación de particiones
      - Formateo lógico
- Dispositivos de almacenamiento terciario
- ...

# Discos

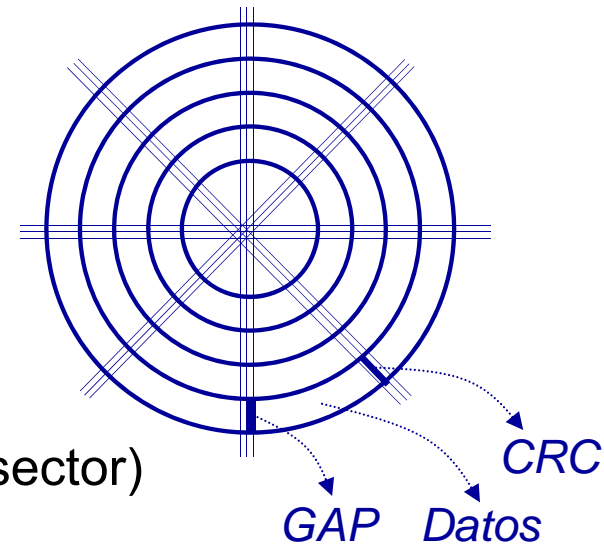
- Utilización:
  - Almacenamiento masivo y no volátil de datos
  - Plataforma para el sistema de intercambio del gestor de memoria virtual
- Taxonomía según la interfaz de su controlador:
  - Dispositivos SCSI (*Small Computer System Interface*)
  - Dispositivos IDE (*Integrated Drive Electronics*)
- Taxonomía según la tecnología de fabricación:
  - Discos duros (*Winchester*) } **Almacenamiento secundario**
  - Discos ópticos (CD-ROM, DVD) } **Almacenamiento terciario**
  - Discos extraíbles (disquetes, ZIP, JAZZ) }

Estructura física y lógica muy similar para todos

# Estructura física de los discos

## ■ Formato y organización de los datos:

- Caras
    - Cilindros (pistas)
    - Sectores / pista
  - Estructura de un sector:
    - ✓ Cabecera (Contiene el número de sector)
    - ✓ Área de datos (256, 512 ó 1024 bytes)
    - ✓ Apéndice (Contiene código de corrección de errores  
Se calcula cada vez que se lee o escribe un sector)
  - Miles de pistas por superficie  
Varios cientos de sectores por pista
- Información usada por el controlador de disco,  
no accesible por usuario*





# Estructura física de los discos

## ■ Características físicas de los sistemas de disco:

- Caras:
  - ✓ Simple cara
  - ✓ Doble cara
- Platos:
  - ✓ Un solo plato
  - ✓ Múltiples platos
- Portabilidad del disco:
  - ✓ Disco no extraíble
  - ✓ Disco extraíble
- Cabezal:
  - ✓ Fijo: Uno por pista
  - ✓ Móvil: Uno por superficie
- Mecanismo del cabezal:
  - ✓ Contacto (disquetes)
  - ✓ Hueco fijo
  - ✓ Hueco aerodinámico (discos Winchester)

# Estructura física de los discos

## ■ Capacidad del disco:

◆  $N^{\circ}$  caras x  $N^{\circ}$  pistas x  $N^{\circ}$  sectores x Tamaño sector

## ■ Tiempo de acceso a disco:

$\Sigma$	(Tiempo de espera	<i>Para acceso al dispositivo)</i>
	Tiempo de posicionamiento	<i>Cabezal bajo la pista de los datos</i>
	Tiempo de latencia	<i>Cabezal bajo el sector de los datos</i>
	Tiempo de transferencia	<i>Entre dispositivo y MC</i>

→ 0 en cabezales fijos

◆ Tiempos medios en discos magnéticos actuales:

- Tiempo de posicionamiento: 5-10 msg.
- Tiempo de latencia: 3-5 msg.

◆ El SO sólo puede influir en el tiempo de posicionamiento



# Estructura física de los discos

- **Apreciaciones a tener en cuenta:**
  - ◆ **Densidad de cada pista:** En los discos modernos el número de sectores varía de unas pistas a otras
  - ◆ **Intercalado de sectores:** La numeración de los sectores no es consecutiva
  - ◆ **Almacenamiento intermedio en el controlador:** Almacenamiento de la pista entera en memoria del controlador
  - ◆ **Controladores inteligentes:** Si controlan varios dispositivos, permiten efectuar operaciones de búsqueda solapadas

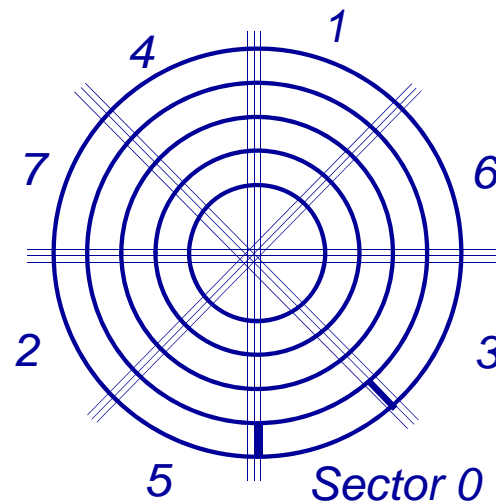


# Estructura física de los discos

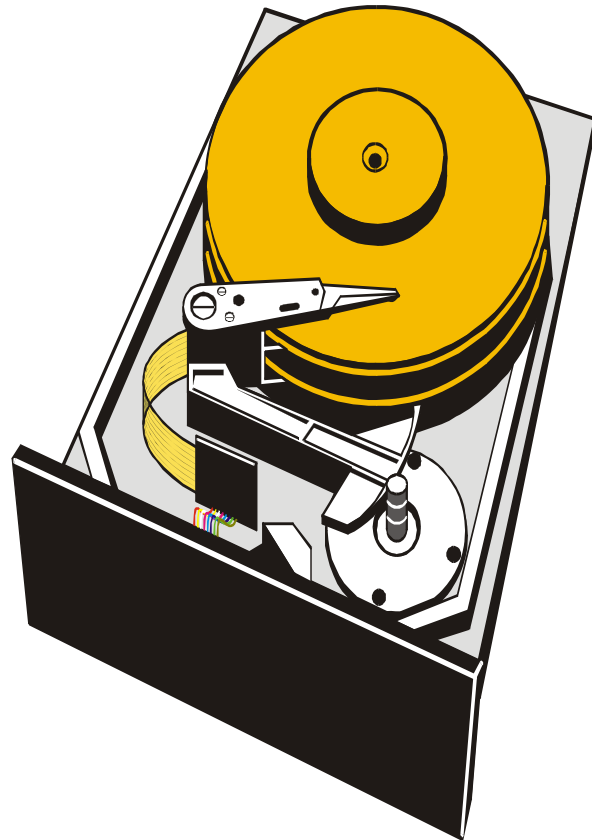
## ■ Apreciaciones a tener en cuenta (cont.):

### ◆ Intercalado de sectores: Ejemplo:

- Sectores por pista = 8
- Factor de entrelazado de bloques = 2



# Ejemplo de disco duro



## SEAGATE Barracuda ATA II ST330630A

Parámetro	Valor
Capacidad:	10.2 GB
Cilindros:	1023
Cabezas:	256
Sectores:	83
Velocidad:	7200 RPM
Tpo. posicionamiento:	8,5 msg.
Tpo. latencia:	4,16 msg.



# Tema 7. Sistemas de entrada/salida

## Índice

- Introducción
- Estructura y componentes del sistema de entrada/salida
- Dispositivos de almacenamiento secundario
  - Discos
    - Estructura física de los discos
    - Formateo de discos
      - Formateo físico
      - Creación de particiones
      - Formateo lógico
- Dispositivos de almacenamiento terciario
- ...





# Formateo físico de discos

## ■ Formateo físico o de bajo nivel:

- ◆ División del disco en sectores para que el controlador de disco pueda leer y escribir
  - Creación de la estructura de datos de cada sector
- ◆ Realizado normalmente como parte del proceso de fabricación
- ◆ Formateo físico de un dispositivo en sistemas UNIX:
  - `fdformat`



# Formateo de discos

## ■ Creación de particiones:

- ◆ Partición o volumen: Porción contigua de sectores del disco
- ◆ Particiones primarias y extendidas
- ◆ Partición activa, de sistema o de arranque
  - Permite arrancar un SO
- ◆ Tabla de particiones:
  - Información sobre la distribución lógica del disco en particiones
  - Almacenada en el sector 0 (primer sector de la primera pista del cilindro más externo)



# Formateo de discos

## ■ Tabla de particiones de un disco:

```
Konsole
Archivo Sessions Opciones Ayuda

[root@cuervo /root]# fdisk /dev/hda

The number of cylinders for this disk is set to 1583.
There is nothing wrong with that, but this is larger than 1024,
and could in certain setups cause problems with:
1) software that runs at boot time (e.g., LILO)
2) booting and partitioning software from other OSs
   (e.g., DOS FDISK, OS/2 FDISK)

Command (m for help): p

Disk /dev/hda: 255 heads, 63 sectors, 1583 cylinders
Units = cylinders of 16065 * 512 bytes

   Device Boot      Start         End      Blocks    Id  System
/dev/hda1  *           1           31     248976    82  Linux swap
/dev/hda2                32        1583    12466440   85  Linux extended
/dev/hda5                32           35         32098+   83  Linux
/dev/hda6                36          557    4192933+   83  Linux
/dev/hda7               558        1583    8241313+   83  Linux

Command (m for help): █
```



# Formateo lógico de discos

- **Formateo lógico de una partición:**
  - ◆ Estructura lógica de un disco: Estructura significativa para el SO
  - ◆ Formateo lógico de una partición → Creación de un SF
    - `format` para sistemas Windows
    - `mkfs` para sistemas Unix



# Formateo lógico de discos

## ■ Tareas de un formateo lógico:

- Construcción del bloque de carga o sector de arranque
- Creación de la estructura lógica del SF
- Creación de una lista de bloques defectuosos
- Creación de una lista de bloques de repuesto





# Formateo lógico de discos

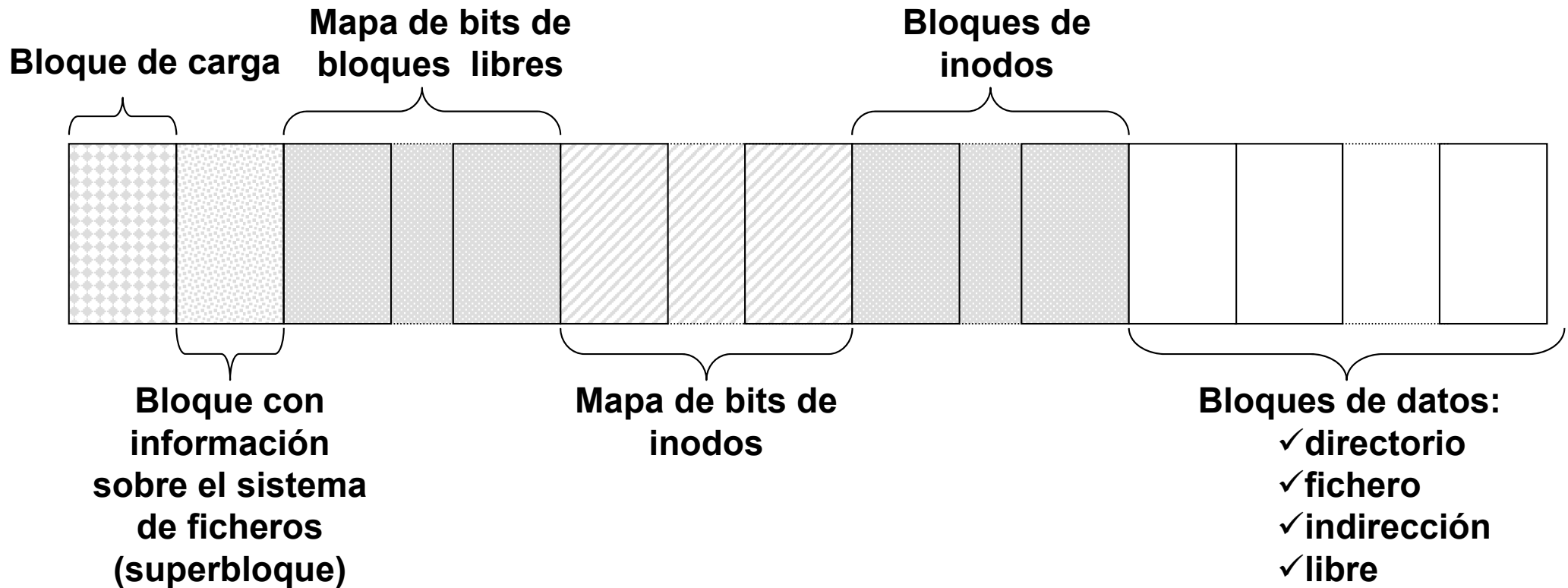
## ■ Bloque de carga o sector de arranque:

- ◆ Contiene información sobre la estructura de la partición y el programa cargador
- ◆ El **programa cargador**:
  - Incluye los aspectos más básicos de iniciación del HW y un bucle de lectura que carga el SO en una dirección de la memoria
  - Cuando termina salta a la dirección de memoria donde puso el núcleo y se arranca el SO

# Formateo lógico de discos

## ■ Creación de la estructura lógica del SF:

- Estructura lógica de un SF con inodos:





# Formateo lógico de discos

## ■ Lista de bloques defectuosos:


- Los bloques en mal estado se marcan como ocupados y no se liberan nunca
- Bloque defectuoso: Cuando alguno de los sectores que lo componen son defectuosos
- Detección por la paridad del sector introducida durante el formato físico

## ■ Lista de bloques de repuesto:

- Permite sustituir un bloque del dispositivo si falla durante el tiempo de vida del mismo

# Tema 7. Sistemas de entrada/salida

## Índice

- Introducción
- Estructura y componentes del sistema de entrada/salida
- Dispositivos de almacenamiento secundario
-  ■ Dispositivos de almacenamiento terciario
- El reloj
- El sistema de E/S en Linux
- El sistema de E/S en Windows NT/2K
- Servicios de E/S en POSIX

# Almacenamiento terciario

## ■ Características:

### ◆ Sistema de almacenamiento con:

- Alta capacidad
- Bajo coste
- Más lentos
- Dispositivos extraíbles donde almacenar datos que no se necesitan inmediatamente en el sistema





# Almacenamiento terciario

## ■ Disquetes, ZIP, JAZZ:

- Pueden ser casi tan rápidos como el disco duro, pero suelen fallar más debido a su constante manipulación

## ■ Discos magneto-ópticos:

- Mayor densidad, capacidad y fiabilidad

## ■ Memorias ópticas y discos digitales versátiles:

- CD, CD-ROM, CD-R, CD-RW
- DVD, DVD-R, DVD-RW

# Almacenamiento terciario

## ■ Cintas magnéticas:

- Soporte de gran capacidad y economía
- Ideal para operaciones que no requieren acceso rápido: respaldos, almacenamiento masivo, etc.
- Manipuladas manualmente o mediante robots que cambian cintas automáticamente
- Los ficheros residentes en disco se pueden archivar en cintas (por razones económicas) y volver a cargar en disco de forma dinámica



# Almacenamiento terciario

## ■ El comando `tar` del intérprete de comandos de sistemas UNIX:


- `tar -cvf /dev/rmt0 /usr/castano` Copia los datos del usuario `castano` al dispositivo de cinta `/dev/rmt0`
- `tar -uvf /dev/rmt0 /usr/castano` Copia anterior incremental
- `tar -tvf /dev/rmt0` Muestra la información almacenada en `/dev/rmt0`
- `tar -xvf /dev/rmt0` Extrae la información almacenada en `/dev/rmt0`

***Copia glogal vs. Copia incremental***



# Tema 7. Sistemas de entrada/salida

## Índice

- Introducción
- Estructura y componentes del sistema de entrada/salida
- Dispositivos de almacenamiento secundario
- Dispositivos de almacenamiento terciario
-  ■ El reloj
- El sistema de E/S en Linux
- El sistema de E/S en Windows NT/2K
- Servicios de E/S en POSIX



# El reloj

## ■ Varias acepciones:

- ◆ Reloj del procesador
- ◆ Reloj del sistema (mantiene fecha y hora)
- ◆ Reloj temporizador (activa periódicamente al SO)

} **Vinculación con SO**

## ■ ¿El reloj es un dispositivo de E/S?

- ◆ Características similares a dispositivos de E/S:
  - Su programación implica usar registros de E/S
  - Usa interrupciones



# HW del reloj

- **Circuito temporizador que genera señal periódica (tic)**
  - ◆ Conectado a línea de interrupción de alta prioridad
  - ◆ Frecuencia programable
- **Reloj CMOS:**
  - ◆ Alimentado por batería que mantiene hora y fecha cuando la máquina está apagada
  - ◆ Consultado por SO en su arranque



# SW del reloj

## ■ Labor principal del SO con el reloj:

- ◆ Gestión de las interrupciones de reloj

## ■ Frecuencia de interrupción:

- ◆ Demasiado alta: Excesiva sobrecarga por tratamiento de interrupciones
- ◆ Demasiado baja: Limitada precisión en medida del tiempo
- ◆ Ejemplo típico: 100 Hz (1 interrupción cada 10 msg.)

## ■ Rutina de interrupción de reloj:

- ◆ Se debe minimizar el trabajo realizado en ella (ya que mientras no se atienden interrupciones de menor prioridad)
- ◆ Solución típica: Dividir trabajo asociado a interrupción de reloj
  - Operaciones más urgentes dentro de la rutina de interrupción
  - Resto: tratamiento posterior fuera de la rutina



# Funciones del manejador de reloj

- **Mantenimiento de fecha y hora**
- **Gestión de temporizadores**
- **Contabilidad y estadísticas**
- **Soporte para la planificación de procesos y recursos**



# Mantenimiento de fecha y hora

## ■ Cálculo de la fecha y hora:

- ◆ En el arranque del SO se lee del reloj CMOS (tras programar la frecuencia del reloj)
- ◆ A partir de entonces el SO actualiza la hora en cada interrupción

## ■ Almacenamiento de fecha y hora:

- ◆ Unidades de tiempo desde fecha fija en el pasado
  - UNIX: sg. o  $\mu$ sg. desde 1-1-1970
  - Windows: centenas de nsg. desde 1-1-1601
- ◆ Diferencias entre horarios de países:
  - El SO almacena la hora en el sistema de tiempo estándar UTC (Universal Coordinated Time)
  - Las bibliotecas se encargan de conversión a horario local

# Gestión de temporizadores

- **¿Quién requiere los temporizadores?**
  - ◆ Los procesos de usuario
  - ◆ El SO crea múltiples temporizadores a partir de temporizador HW
    - Módulo de comunicaciones
    - Manejador de disquete, etc.
- **Implementación de temporizadores:**
  - ◆ Lista de temporizadores activos
    - Elemento de lista: Plazo restante (en tics) + Función a ejecutar al cumplir
- **Gestión de temporizadores:**
  - ◆ Puede consumir bastante tiempo
    - Típicamente realizada fuera de rutina de interrupción

- La ejecución periódica de la rutina de interrupción de reloj permite muestrear diversos aspectos del estado del sistema
  - ◆ Contabilidad de uso de procesador de cada proceso
    - Se le carga el uso de procesador al proceso en ejecución
    - Se distingue si estaba en modo usuario o sistema






# Soporte para la planificación

## ■ Algoritmos de planificación:

- ◆ En la mayoría influye el tiempo  $\Rightarrow$  La rutina de interrupción de reloj conlleva acciones de planificación
- ◆ Round-Robin:
  - En cada interrupción se descuenta tiempo a la rodaja
  - Cuando llega a cero  $\rightarrow$  Replanificación
- ◆ Algoritmos de prioridad:
  - Pueden requerir recalcular periódicamente prioridad

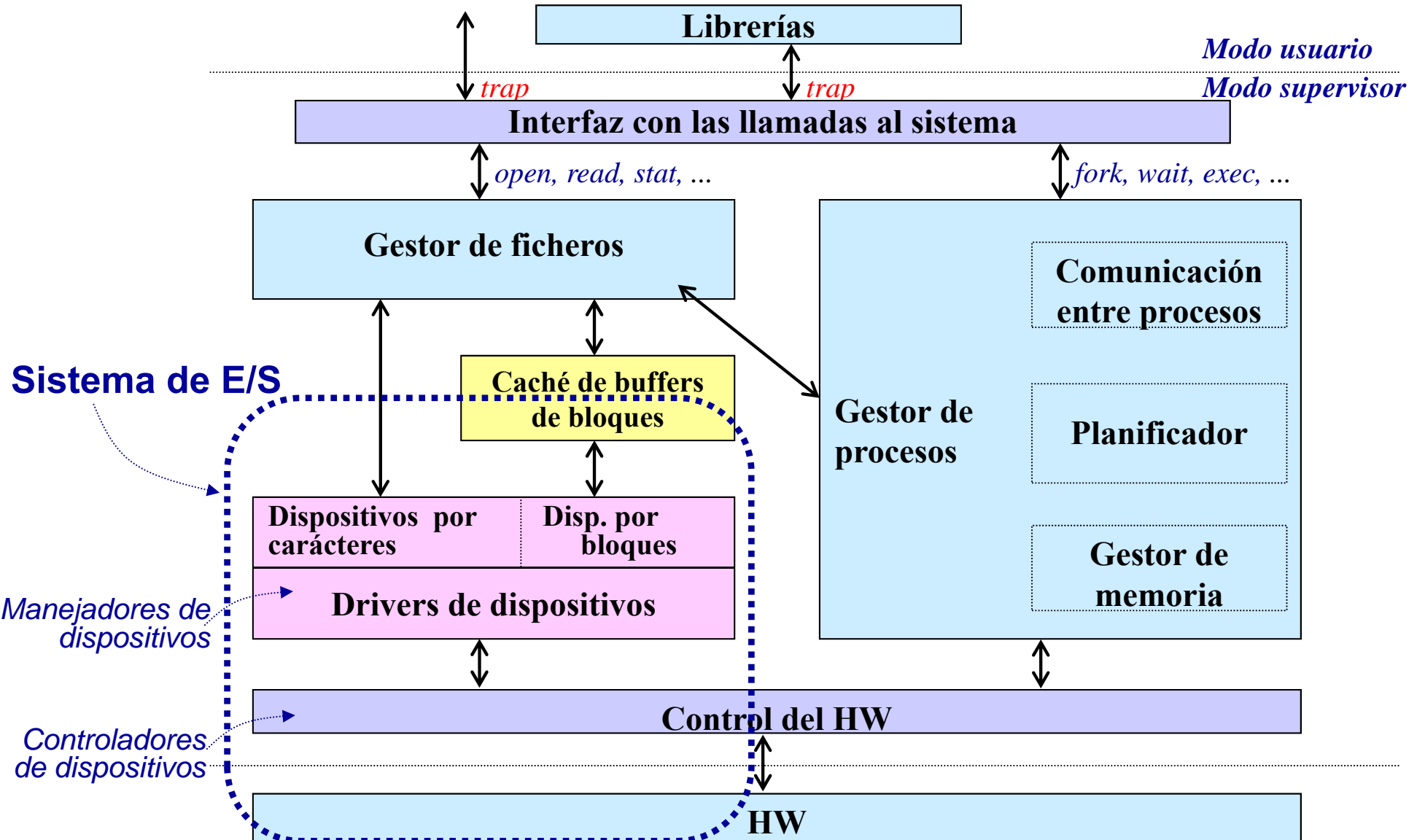
# Tema 7. Sistemas de entrada/salida

## Índice

- Introducción
- Estructura y componentes del sistema de entrada/salida
- Dispositivos de almacenamiento secundario
- Dispositivos de almacenamiento terciario
- El reloj
-  ■ El sistema de E/S en Linux
- El sistema de E/S en Windows NT/2K
- Servicios de E/S en POSIX




# Diagrama de bloques del núcleo de Linux



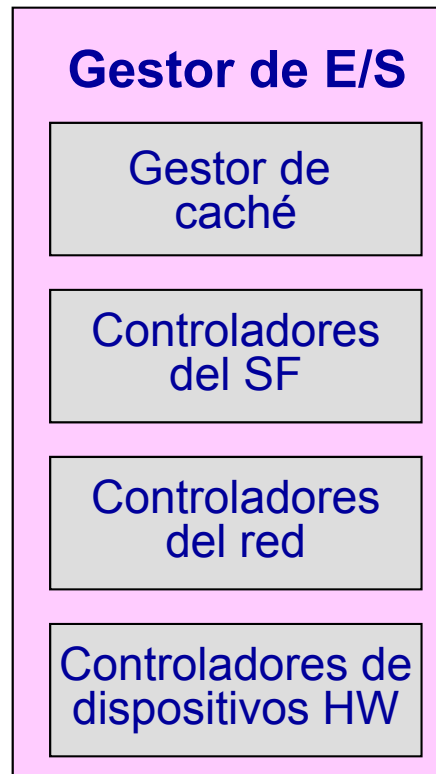
# Tema 7. Sistemas de entrada/salida

## Índice

- Introducción
- Estructura y componentes del sistema de entrada/salida
- Dispositivos de almacenamiento secundario
- Dispositivos de almacenamiento terciario
- El reloj
- El sistema de E/S en Linux
-  ■ El sistema de E/S en Windows NT/2K
- Servicios de E/S en POSIX

# E/S en Windows NT/2K

## ■ Módulos básicos de E/S:



# Tema 7. Sistemas de entrada/salida

## Índice

- Introducción
- Estructura y componentes del sistema de entrada/salida
- Dispositivos de almacenamiento secundario
- Dispositivos de almacenamiento terciario
- El reloj
- El sistema de E/S en Linux
- El sistema de E/S en Windows NT/2K
- Servicios de E/S en POSIX



# Servicios de E/S en POSIX

- **Servicios relacionados con el reloj:**
  - ◆ Servicios de hora y fecha
    - Para leer y cambiar (sólo superusuario) fecha y hora
  - ◆ Temporizaciones
  - ◆ Servicios de contabilidad
    - Ejemplo: tiempo de procesador consumido por un proceso
- **Servicios de entrada/salida:**
  - ◆ Acceso a dispositivos a través de interfaz de archivos

# Servicios de E/S en POSIX

## ■ Lectura de fecha y hora:

### ◆ Sintaxis:

```
time_t time (time_t *t);
```

### ◆ Descripción:

- Devuelve número de segundos desde 1-1-1970

### ◆ Funciones de biblioteca para convertir a año, mes, ..., segundos:

- `localtime`



# Servicios de E/S en POSIX

## ■ Ejemplo:

```
#include <stdio.h>
#include <time.h>

int main()
{
    time_t tiempo;
    struct tm *fecha;

    tiempo=time(NULL);
    fecha=localtime(&tiempo);

    printf("%02d/%02d/%04d %02d:%02d:%02d\n",
           fecha->tm_mday, fecha->tm_mon, fecha->tm_year+1900,
           fecha->tm_hour, fecha->tm_min, fecha->tm_sec);

    exit(0);
}
```

# Servicios de E/S en POSIX

## ■ Modificación de fecha y hora:

### ◆ Sintaxis:

```
int stime (time_t *t);
```

### ◆ Descripción:

- Fija hora según parámetro (segundos desde 1-1-1970)
- Sólo para superusuario

# Servicios de E/S en POSIX

## ■ Activación de una alarma:

### ◆ Sintaxis:

```
unsigned int alarm (unsigned int seconds);
```

### ◆ Descripción:

- Envía al proceso que la invoca la señal SIGALRM tras seconds segundos

# Servicios de E/S en POSIX

## ■ Servicios de contabilidad:

### ◆ Sintaxis:

```
clock_t times (struct tms *info);
```

### ◆ Descripción:

- Proporciona a partir de info información sobre tiempo de ejecución de proceso e hijos
- Devuelve valor relacionado con tiempo real en el sistema  
Típicamente *ticks* de reloj desde arranque del sistema
- Este valor no se usa de forma absoluta  
Se realizan dos llamadas y se calcula la diferencia



# Servicios de E/S en POSIX

## ■ Ejemplo:

```
#include <stdio.h>
#include <unistd.h>
#include <time.h>
#include <sys/times.h>

int main(int argc, char *argv[]){
    struct tms InfoInicio, InfoFin;
    clock_t t_inicio, t_fin;
    long tickporseg;

    tickporseg=sysconf(_SC_CLK_TCK); /* Num interrupciones reloj por seg */
    t_inicio=times(&InfoInicio);
    if (fork()==0) {execvp (argv[1], &argv[1]); perror("Error hijo"); exit(-1);}
    wait(NULL);
    t_fin=times(&InfoFin);

    printf("Tiempo real: %7.2f\n", (float)(t_fin - t_inicio)/tickporseg);
    printf("Tiempo de usuario: %7.2f\n",
           (float)(InfoFin.tms_cutime-InfoInicio.tms_cutime)/tickporseg);
    printf("Tiempo de sistema: %7.2f\n",
           (float)(InfoFin.tms_cstime-InfoInicio.tms_cstime)/tickporseg);

    exit(0);
}
```



# Servicios de E/S en POSIX

## ■ Servicios de E/S sobre dispositivos:

### ◆ Sintaxis:

```
int ioctl (int fildes, int request, arg);
```

### ◆ Descripción:

- Realiza sobre el dispositivo asociado a `fildes` la operación especificada en `request` para la que se necesitarán los parámetros especificados a través de `arg`
- No definida en POSIX

*Tomarán unos valores u otros según el tipo de fichero*