

TEMA 1. CARACTERIZACIÓN DE SISTEMAS OPERATIVOS

Implantación de Sistemas Operativos

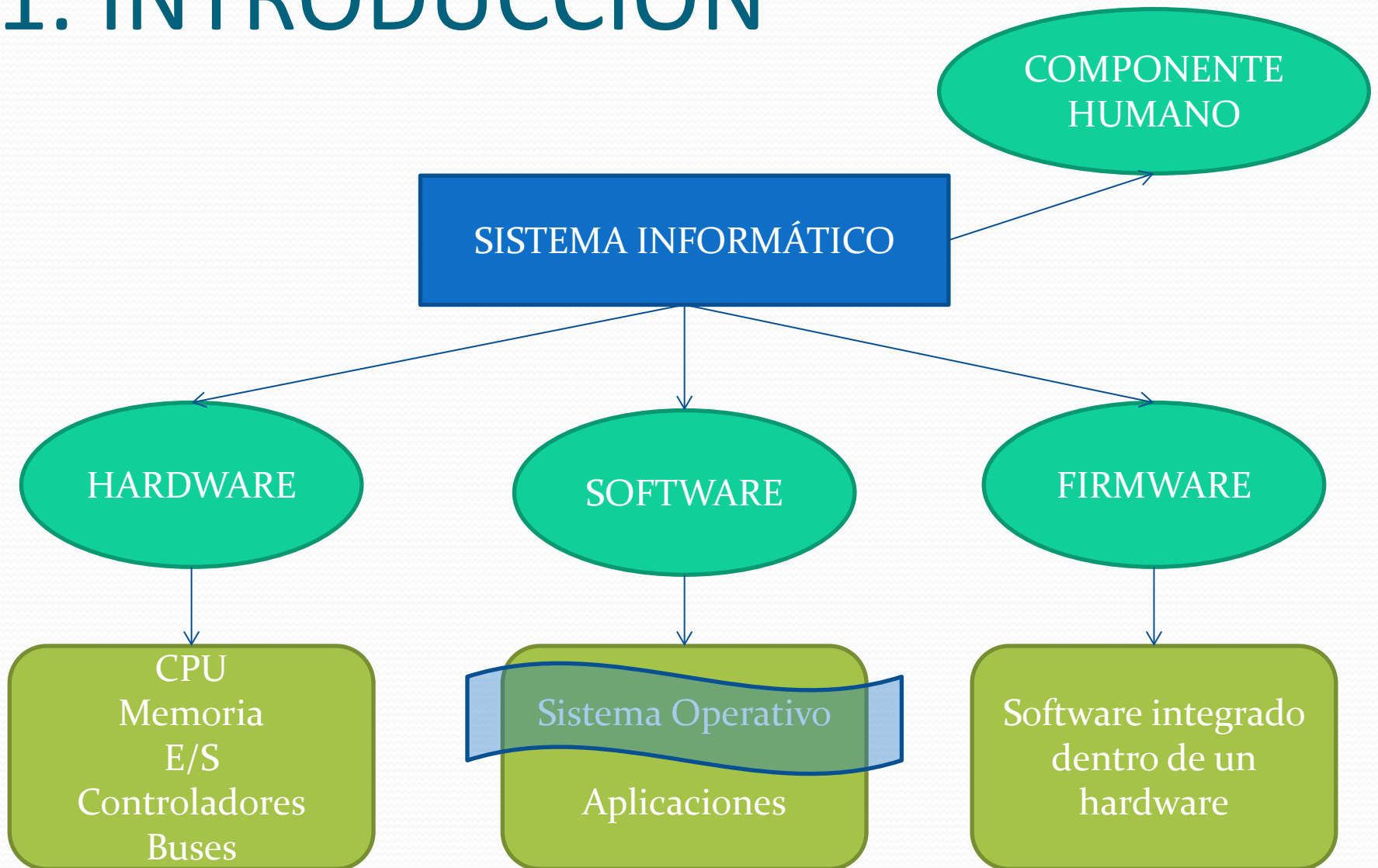
1º ASIR

Profesora: Anabel Serradilla

CONTENIDOS

1. Introducción
2. Sistemas Operativos
3. Funciones de los SS.OO.
4. Evolución histórica
5. Clasificación
6. Tipos de licencias
7. Ejemplos de SSOO
8. Gestión de procesos
9. Gestión de memoria
10. Gestión de E/S
11. Gestión del sistema de archivos
12. Gestión de la red
13. Protección y Seguridad
14. Gestores de arranque
15. Tendencias actuales

1. INTRODUCCIÓN



2. SISTEMAS OPERATIVOS

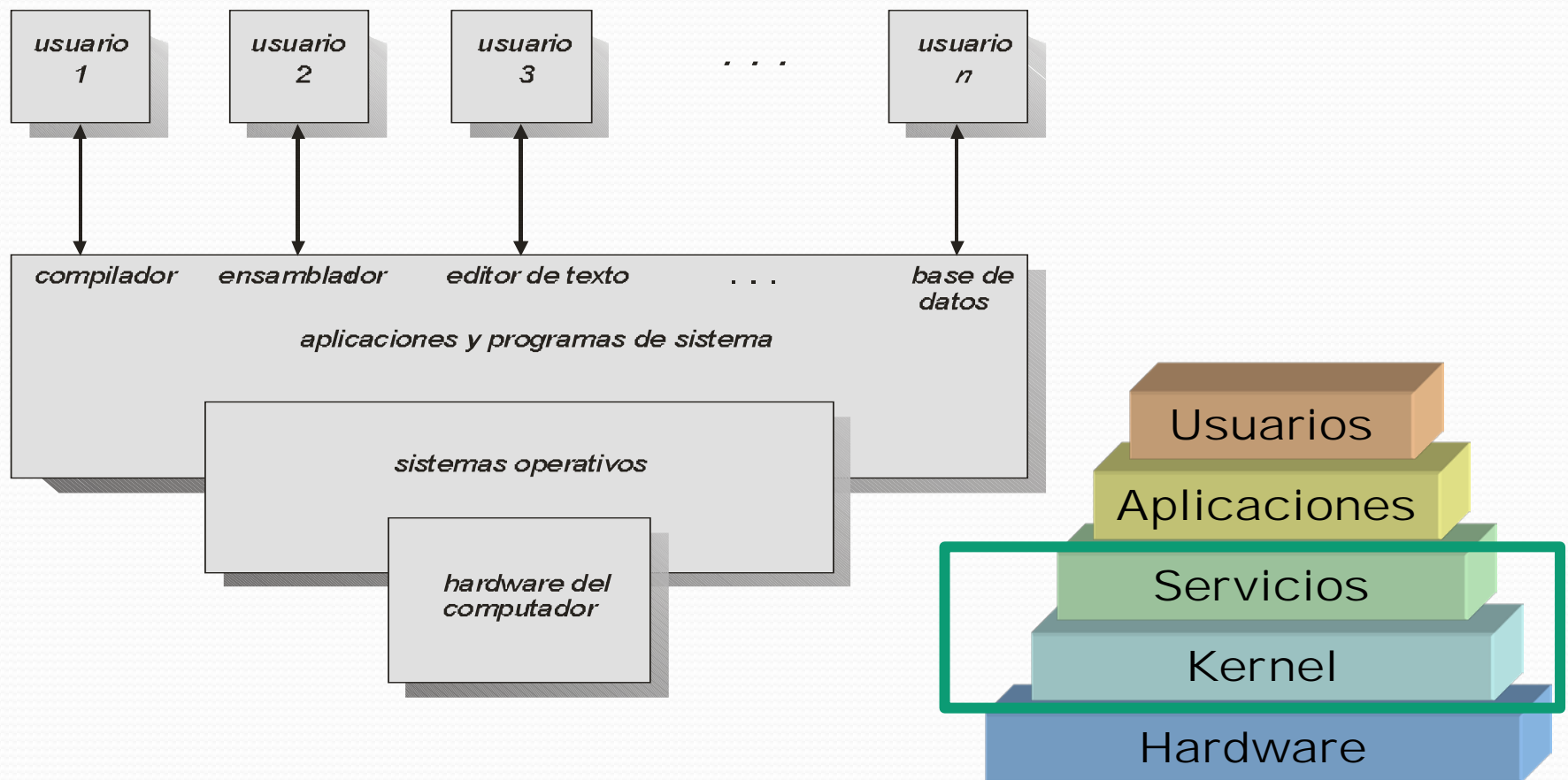
Definición

Programa que actúa como intermediario entre el usuario del sistema informático y el hardware.

Permite ocultar los detalles del hardware, gestionar los recursos del sistema y optimizar su uso



2. SISTEMAS OPERATIVOS



3. FUNCIONES DE LOS SS.OO.

- Facilitar el uso al usuario, independizando los programas de aplicación del hardware
- Ejecutar programas de usuario
- Acceder a los dispositivos de entrada / salida (periféricos)
- Gestionar las comunicaciones en red
- Permitir a los usuarios compartir recursos

3. FUNCIONES DE LOS SS.OO.

- Administrar la memoria
- Proporcionar una estructura y conjunto de operaciones para el sistema de archivos
- Proporcionar interfaces de usuario: en modo texto y gráficos
- Servicios soporte: actualizaciones de software, controladores para nuevos periféricos, etc.

3. FUNCIONES DE LOS SS.OO.

- Gestión de procesos
- Gestión de memoria
- Gestión de E/S
- Gestión de ficheros
- Gestión de la red
- Protección y seguridad

3. FUNCIONES DE LOS SS.OO.

¿Qué tareas nos evita un Sistema Operativo?

- Al guardar un fichero en disco duro, la localización en memoria no se hace por “Cilindros-Cabeza-Sector”, sino por ficheros (sistemas de ficheros FAT, FAT32, NTFS, EXT)
- Para navegar por Internet no necesitamos dividir la información en paquetes y asignarles puertos, IP, MAC; el navegador y el S.O. nos evitan esa tarea
- Al conectar un USB al ordenador no debemos buscar un controlador para él. El S.O. gestiona los dispositivos de I/O

4. EVOLUCIÓN HISTÓRICA

- ENIAC y primeros ordenadores: no disponen de S.O.
- 1970-1980: Primer S.O. de propósito general → UNIX
- A partir de 1980:
 - MS-DOS para el IBM-PC y MacOS para Apple Computer
 - Interfaz de ventanas → MS Windows
 - Linux
 - Otros dispositivos → Android, Windows Phone, Apple iOS

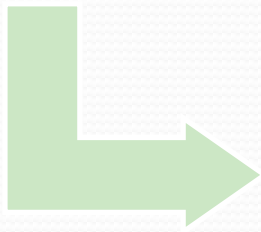
4. EVOLUCIÓN HISTÓRICA



4. EVOLUCIÓN HISTÓRICA

Usuarios con
tiempo asignado

- Tareas sin terminar
- Tiempo de proceso desaprovechado



Operador

- Ejecutan rutinas de carga y descarga
- Agrupan trabajos



Monitor
residente

- Programa en memoria
- Lenguaje de control de trabajos

4. EVOLUCIÓN HISTÓRICA

Formas de trabajo

- Proceso en serie
 - Primeros ordenadores sin sistema operativo
 - Programador accede al hardware directamente
 - Controla el sistema con indicadores luminosos
 - Código máquina y salida por impresora
 - Error → indicadores luminosos
 - No error → salida por impresora
 - Serie: un trabajo detrás de otro
 - Problemas: planificación y tiempo de preparación

4. EVOLUCIÓN HISTÓRICA

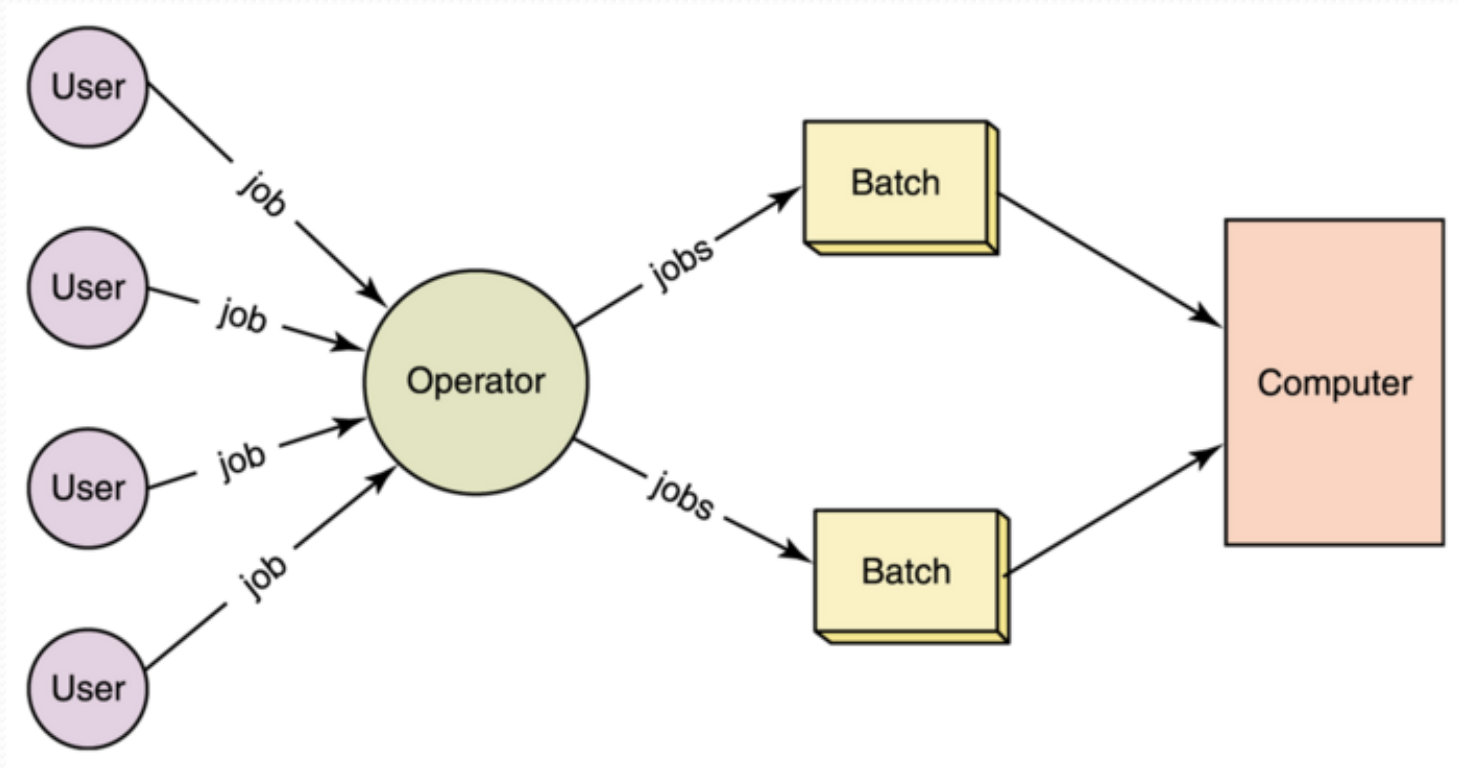
Formas de trabajo

- Sistemas por lotes (batch)
 - Se agrupan los trabajos por lotes
 - Monitor → software que controla la ejecución de los trabajos. Planifica su ejecución y los prepara.
 - El procesador ejecuta cada trabajo hasta que termina o detecta un error → control al monitor
 - Protección de memoria usada por el monitor
 - Uso de temporizador para repartir los tiempos
 - Instrucciones privilegiadas, solo para monitor
 - Tiempo de máquina para procesos y monitor

4. EVOLUCIÓN HISTÓRICA

Formas de trabajo

- Sistemas por lotes (batch)



4. EVOLUCIÓN HISTÓRICA

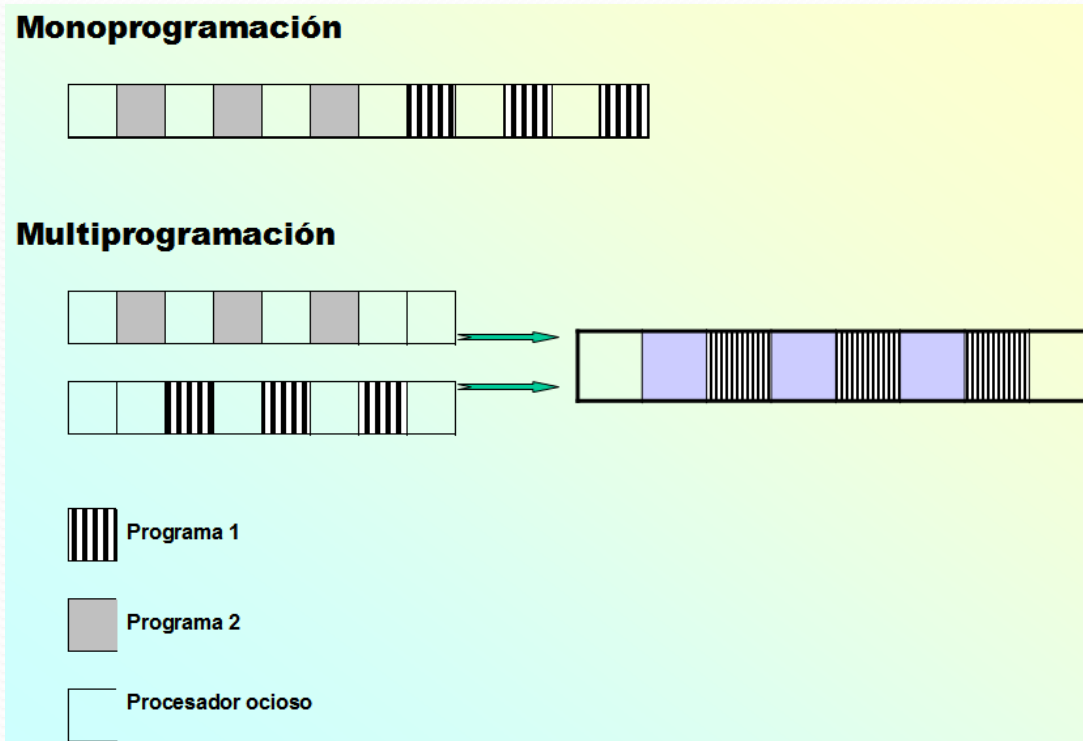
Formas de trabajo

- Sistemas por lotes con multiprogramación
 - Operaciones de E/S lentas → el procesador está mucho tiempo desocupado.
 - Se aprovecha este tiempo cargando varios programas en memoria y alternando unos con otros
 - Proceso conocido como multiprogramación o multitarea

4. EVOLUCIÓN HISTÓRICA

Formas de trabajo

- Sistemas por lotes con multiprogramación



4. EVOLUCIÓN HISTÓRICA

Formas de trabajo

- Sistemas de tiempo compartido
 - A veces es necesario que el usuario interactúe con el computador.
 - Se alternan tareas por lotes con tareas interactivas
 - El tiempo se comparte entre los distintos usuarios del sistema

4. EVOLUCIÓN HISTÓRICA

Formas de trabajo

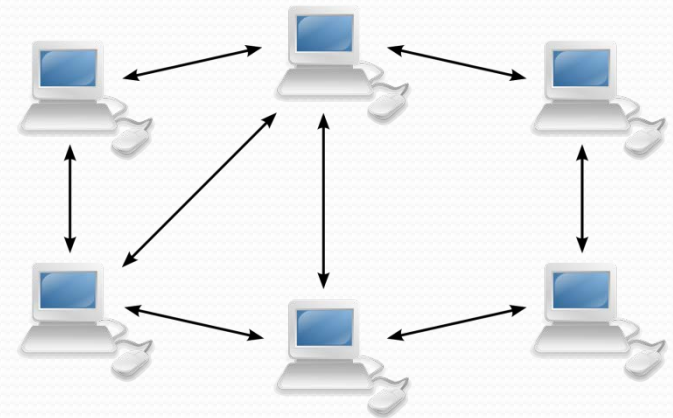
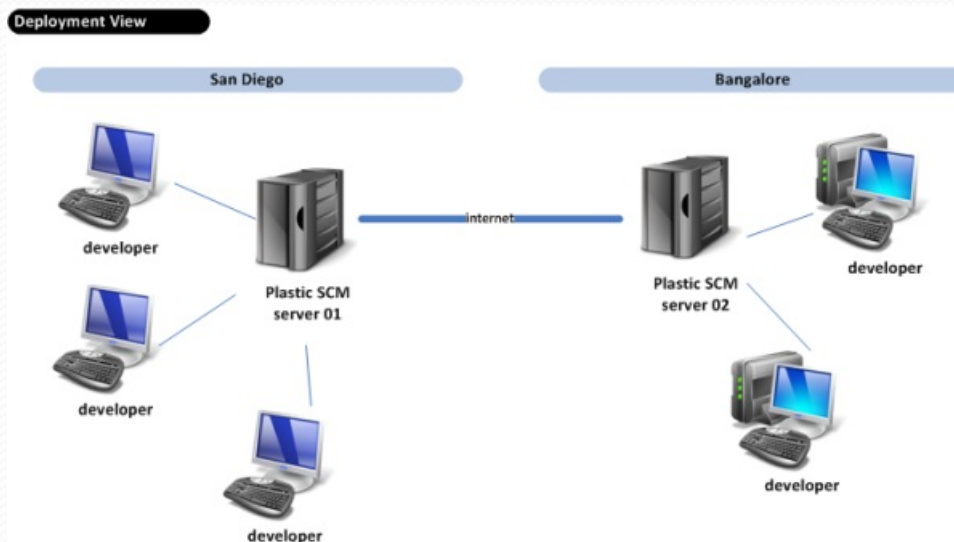
- Sistemas distribuidos
 - Procesamiento centralizado → todas las funciones están centralizadas en una CPU.
 - Procesamiento distribuido → los datos se ejecutan en distintos nodos dispersos geográficamente y conectados en red.
 - Fragmentación de los elementos que componen una aplicación
 - Recursos controlados de forma independiente
 - Diferentes formas: cliente/servidor o punto a punto

4. EVOLUCIÓN HISTÓRICA

Formas de trabajo

■ Sistemas distribuidos

Cliente / Servidor



Punto a punto

5. CLASIFICACIÓN

Por su estructura

Sistemas operativo monolíticos

Sistemas operativos jerárquicos

Máquina virtual

Microkernel o Cliente - Servidor

5. CLASIFICACIÓN

- **Sistemas operativos monolíticos:** es un único programa el que gestiona todo en modo supervisor: planificación de procesos, administración de memoria principal y ficheros y gestión de E/S.

Ventaja: rendimiento

Inconveniente: un cambio requiere recompilación y reinicio del sistema.

5. CLASIFICACIÓN

- **Sistemas operativos jerárquicos:** el S.O. está organizado en capas diferenciadas por la función que realizan.

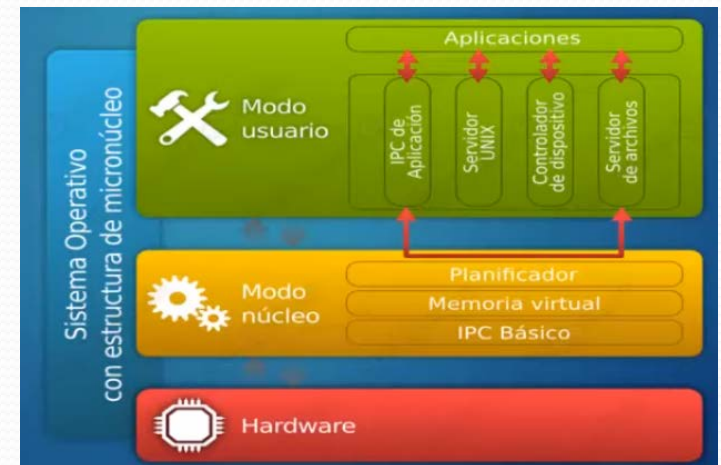
Ventaja: facilidad de construcción,
independencia de las capas.

Ejemplo: Multics



5. CLASIFICACIÓN

- **Máquina virtual:** implementación por software de una máquina que ejecuta instrucciones como si fuera una máquina física.
- **Microkernel o Cliente-Servidor:** las distintas tareas están distribuidas en porciones de código modulares y sencillas

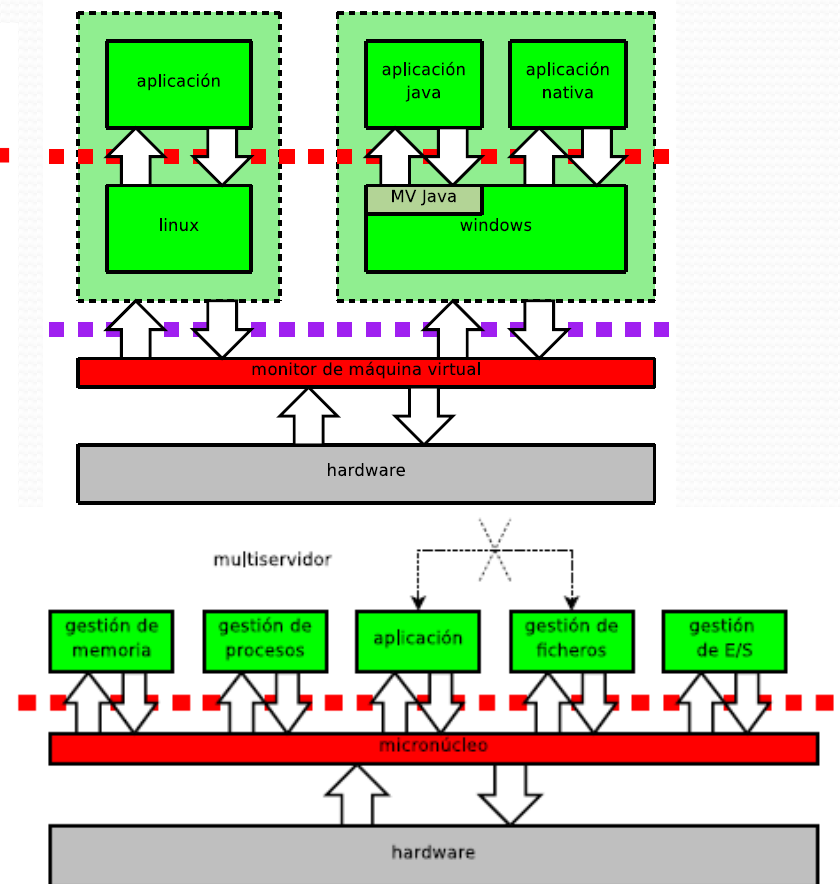
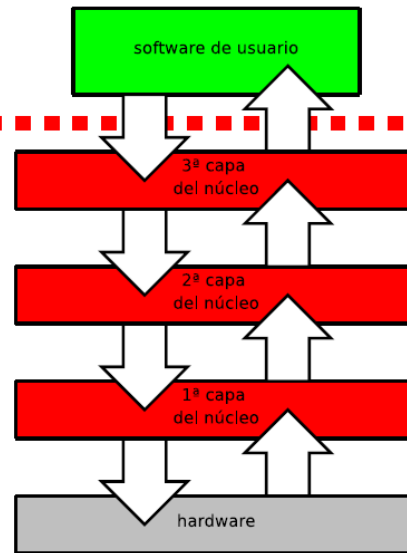
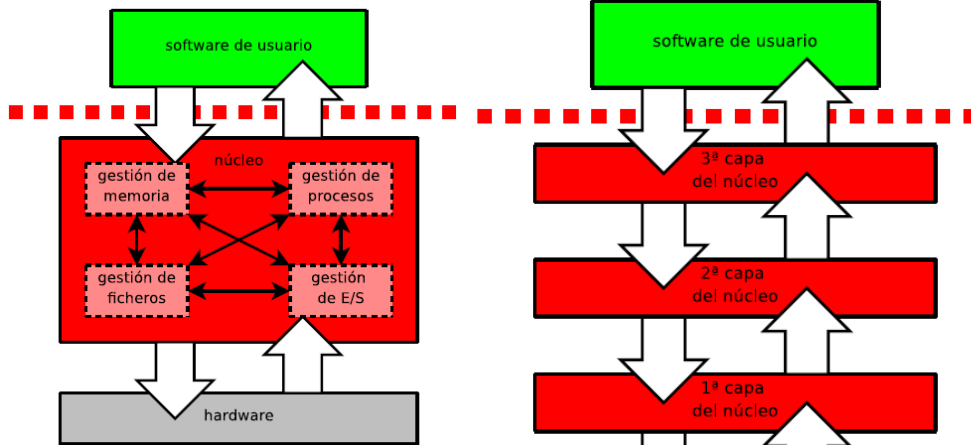


5. CLASIFICACIÓN

Monolítica

Jerárquica

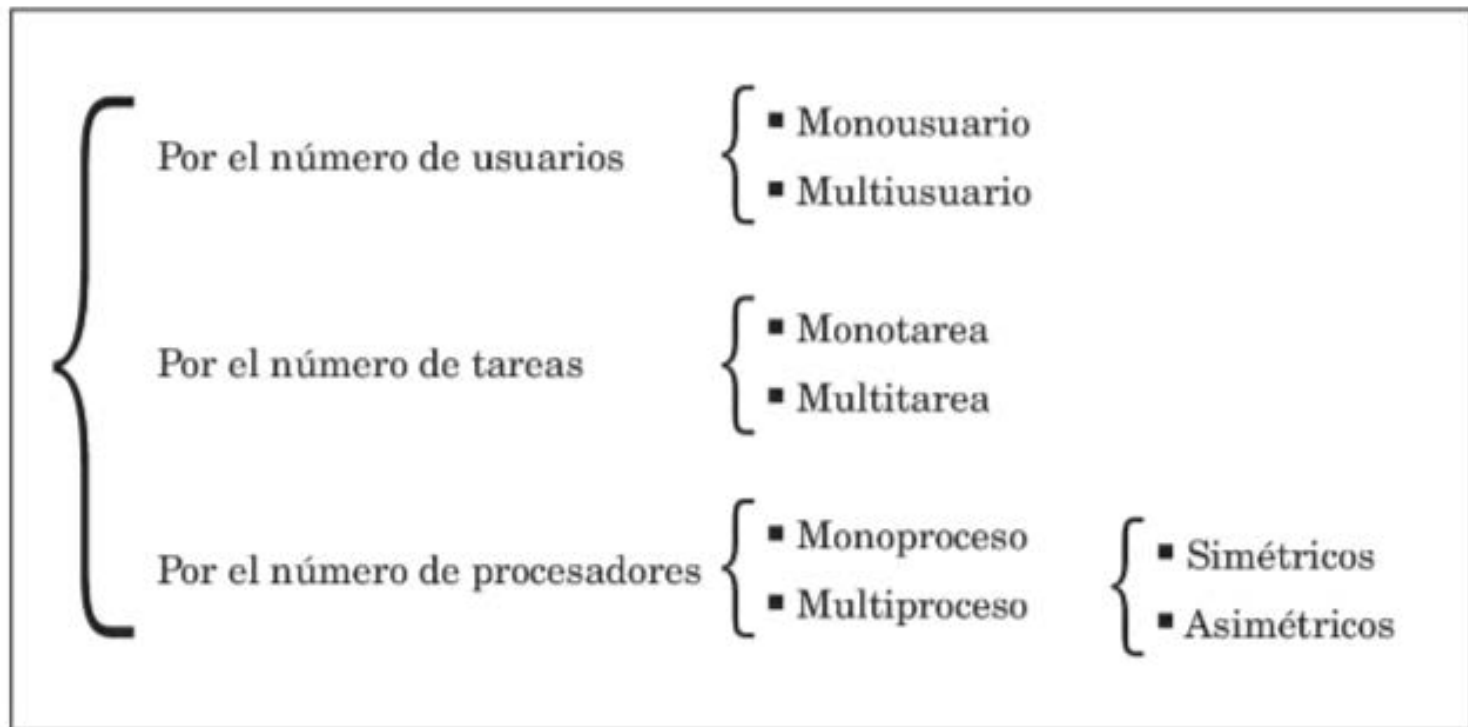
Máquina virtual



Cliente-Servidor

5. CLASIFICACIÓN

Por los servicios ofrecidos



5. CLASIFICACIÓN

Por cómo se ofrecen los servicios



Sistemas centralizados

Sistemas distribuidos

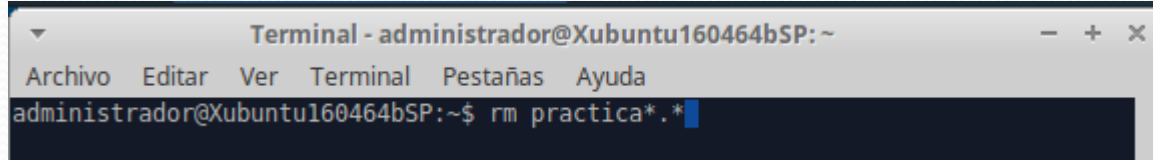
Sistemas operativos en red

Sistemas operativos de escritorio

5. CLASIFICACIÓN

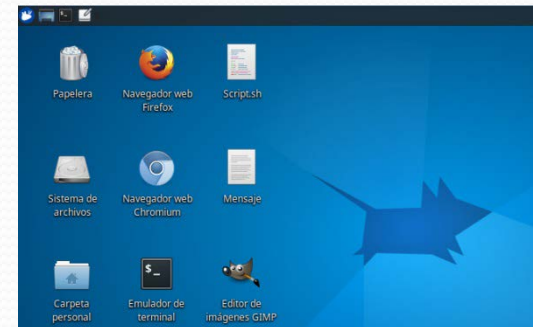
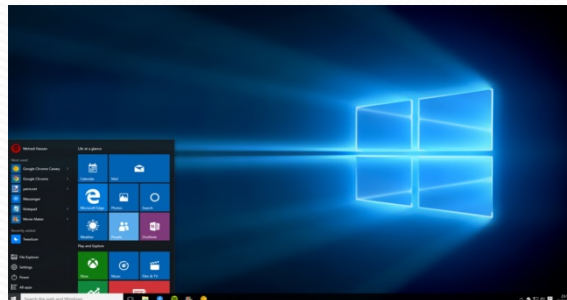
Por su forma de uso

Modo comando

A screenshot of a terminal window titled "Terminal - administrador@Xubuntu160464bSP: ~". The window has a menu bar with "Archivo", "Editar", "Ver", "Terminal", "Pestañas", and "Ayuda". The command prompt shows "administrador@Xubuntu160464bSP:~\$ rm practica*.*" with a blue cursor at the end.

```
Terminal - administrador@Xubuntu160464bSP: ~
Archivo  Editar  Ver  Terminal  Pestañas  Ayuda
administrador@Xubuntu160464bSP:~$ rm practica*.*
```

Interfaz gráfica



5. CLASIFICACIÓN

Por disponibilidad



Sistemas operativos propietarios

Sistemas operativos libres

6. TIPOS DE LICENCIAS

El software propietario puede ser:

OEM: su venta está vinculada a la compra de un equipo nuevo. No se puede comprar de forma independiente.

Retail: venta de software. La propiedad del software es del comprador.

Licencias por volumen: software cuyas licencias se venden en paquetes a las empresas, p.e. 25 licencias

6. TIPOS DE LICENCIAS

Otros conceptos

Código abierto (Open source): se distribuye el código fuente del software.

Software libre: software de código abierto que permite la copia, la distribución y la modificación sin restricciones.

FSF (Free Software Foundation): organización sin ánimo de lucro que promueve los ideales del software libre.

6. TIPOS DE LICENCIAS

Licencias del software libre

Licencias GPL: el autor conserva los derechos de autor. Las modificaciones del software deben cumplir también con los requisitos GPL.

Licencias BSD: el autor solo conserva algunos derechos. Compatible con GPL.

Copyleft: los términos de distribución no permiten a los redistribuidores agregar ninguna restricción adicional cuando lo redistribuyen o modifican, o sea, la versión modificada debe ser también libre.

6. TIPOS DE LICENCIAS

Comparativa

Licencia EULA sistema operativo Windows XP	Licencia GPL sistema operativo Linux
Se prohíbe la copia.	Permite la copia, modificación y redistribución del software, ya que tiene el código fuente.
Es puede utilizar con un único ordenador con un máximo de dos procesadores.	Se puede vender y se pueden cobrar servicios sobre el software.
No se puede utilizar como servidor web (<i>web server</i>) ni como <i>file severo</i> .	Cualquier patente sobre el software debe ser licenciada por el beneficio de todos.
Hay que registrarse en los treinta días de utilización.	El software modificado no debe tener coste para la licencia.
La licencia puede dejar de ser válida si se efectúan cambios en el hardware.	no
Si la compañía quiere, las actualizaciones del sistema pueden modificar la licencia.	no
Sólo se puede transferir una vez a otro usuario.	Se puede enviar a muchos usuarios.
Impone una limitación sobre la ingeniería inversa.	no
Da derecho a Microsoft para que en cualquier momento pueda recoger información sobre el sistema y su utilización y para que entregue esta información a terceros.	no
La garantía es para los primeros noventa días.	No ofrece garantía.
Las actualizaciones y los parches no tienen garantía.	Proporciona garantía de los derechos del usuario a la copia, modificación y redistribución del software.
Coste para obtener la licencia.	Licencia gratuita.

7. EJEMPLOS DE SS.OO.

UNIX

- Nació en los años 70
- Sistema multiusuario y multitarea
- Procede del antiguo sistema MULTICS y se desarrolló en lenguaje C
- En su momento no se pudo comercializar y se ofreció de forma gratuita a las universidades para la investigación, lo que fomentó su desarrollo
- Fue la base para otros sistemas, tanto gratuitos como de pago

7. EJEMPLOS DE SS.OO.

LINUX

- Fue creado en 1991 por Linus Torvalds
- El objetivo era crear una versión de UNIX que funcionase sobre cualquier PC compatible
- Software libre desarrollado por un grupo de colaboradores que trabajan de forma desinteresada
- El kernel de Linux y las aplicaciones GNU se unieron dando lugar al SO GNU/Linux
- Se pueden descargar por separado (kernel, entorno gráfico, aplicaciones GNU, sistema de ventanas, ...) o en una distribución completa.

7. EJEMPLOS DE SS.OO.

LINUX - Distribuciones

Existen más de 500 distribuciones de Linux diferentes.

Las más importantes son:

- Debian: Ubuntu, Xubuntu, Linux Mint
- Red Hat: Fedora
- Slackware: SUSE, openSUSE



7. EJEMPLOS DE SS.OO.

WINDOWS

- Su primera versión (Windows 1.0) es del año 1985
- Versiones:



Windows 3.1 (1992)

Windows NT (1993)

Windows 95 (1995)

Windows 98 (1998)

Windows Me (2000)

Windows XP (2001)



Windows Vista (2006)

Windows 7 (2009)

Windows 8 (2012)

Windows 10 (2015)

Windows 11 (2021)



7. EJEMPLOS DE SS.OO.

Mac OS

- Creado por Apple para ordenadores Macintosh
- Primer S.O. con interfaz gráfica (1984)
- A partir de Mac OS X, el S.O. se deriva de UNIX pero manteniendo la interfaz gráfica
- Versiones: Mac OS 7
Mac OS 8
Mac OS 9
Mac OS X

7. EJEMPLOS DE SS.OO.

Otros sistemas operativos

Para otros dispositivos diferentes de PCs como smartphones, tablets, ...

- Android, versión modificada del kernel Linux
- iOS, para iPhones o iPads
- Windows Phone, desarrollado por Microsoft
- Google Chrome OS, basado en el kernel de Linux. Diseñado especialmente para los chromebooks.

8. GESTIÓN DE PROCESOS

Concepto de proceso

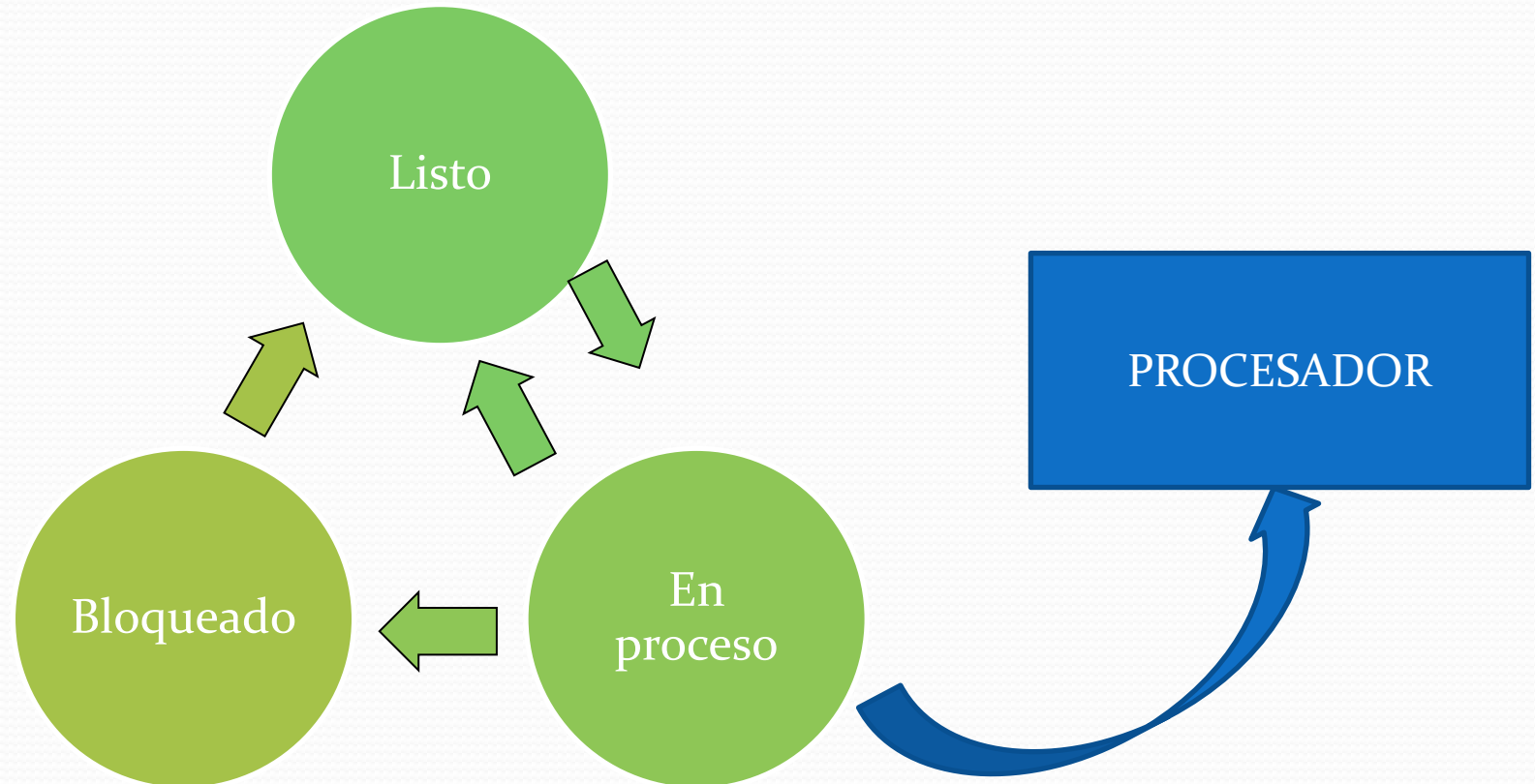
Un proceso es un programa en ejecución.

En un instante dado un proceso puede estar en uno de estos tres estados:

- **Listo:** son los que pueden pasar a estado de ejecución si el planificador del S.O. lo selecciona
- **En ejecución:** son los que están ejecutándose en el procesador en un momento dado
- **Bloqueado:** son los que están esperando respuesta de algún otro proceso para poder continuar

8. GESTIÓN DE PROCESOS

Concepto de proceso

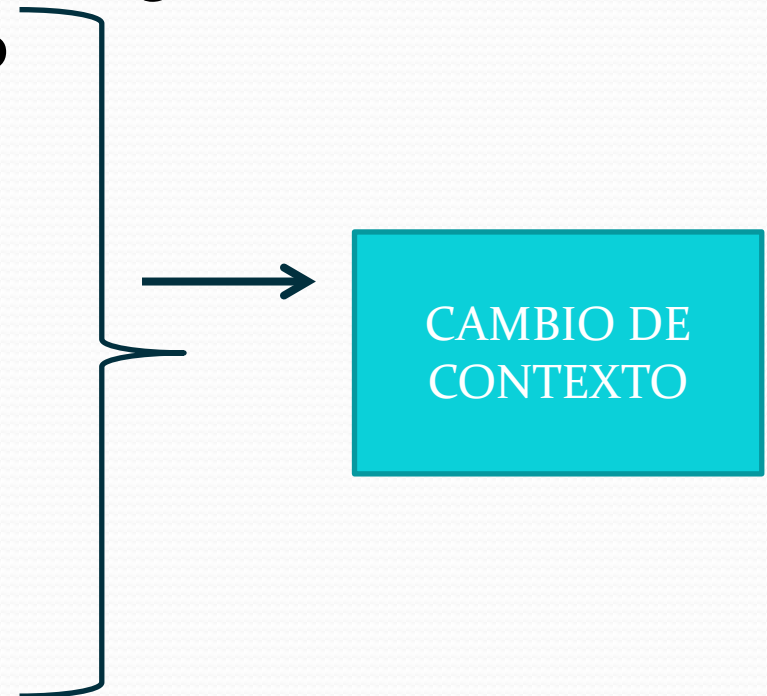


8. GESTIÓN DE PROCESOS

Concepto de proceso

Un proceso tiene asociados los siguientes elementos:

- Identificador del proceso
- Estado
- Contador de programa
- Valores de los registros
- Espacio en memoria
- Prioridad
- Propietario
- ...



8. GESTIÓN DE PROCESOS

Planificación del procesador

En la planificación del procesador se decide cuánto tiempo de ejecución se le asigna a cada proceso y en qué momento

Una estrategia de planificación debe buscar que los procesos tengan sus turnos de ejecución de forma apropiada, con un buen rendimiento y sin sobrecargar el planificador

8. GESTIÓN DE PROCESOS

Planificación del procesador

- Planificación no apropiativa: cuando a un proceso le toca su turno de ejecución, ya no puede ser suspendido.
- Planificación apropiativa: el S.O. puede arrebatarse el uso de la CPU a un proceso que esté ejecutándose. Hay un reloj generando interrupciones en las cuales el planificador toma el control y decide qué proceso debe ejecutarse.

Algoritmos: Round Robin, prioridades, el más corto, ...

8. GESTIÓN DE PROCESOS

Planificación del procesador

PROCESO A

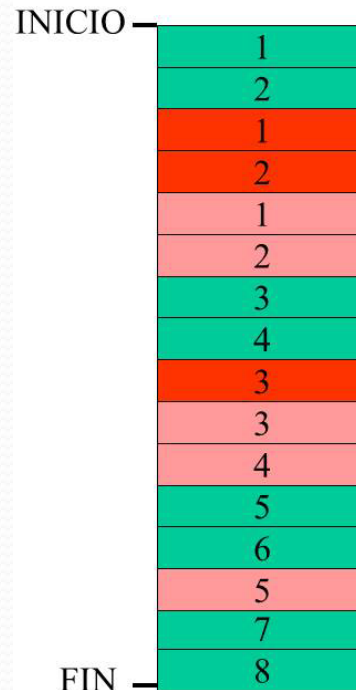
1
2
3
4
5
6
7
8

PROCESO B

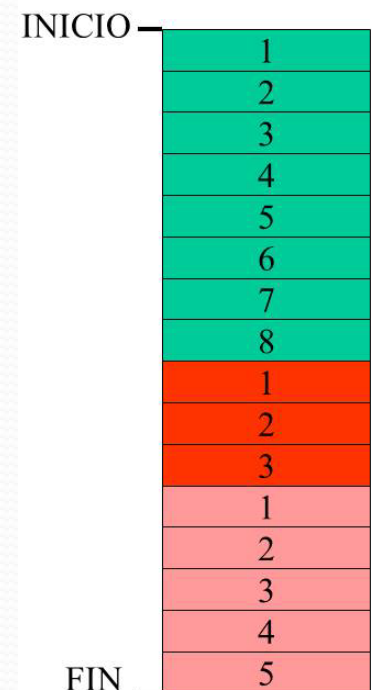
1
2
3
4
5

PROCESO C

1
2
3



Round Robin

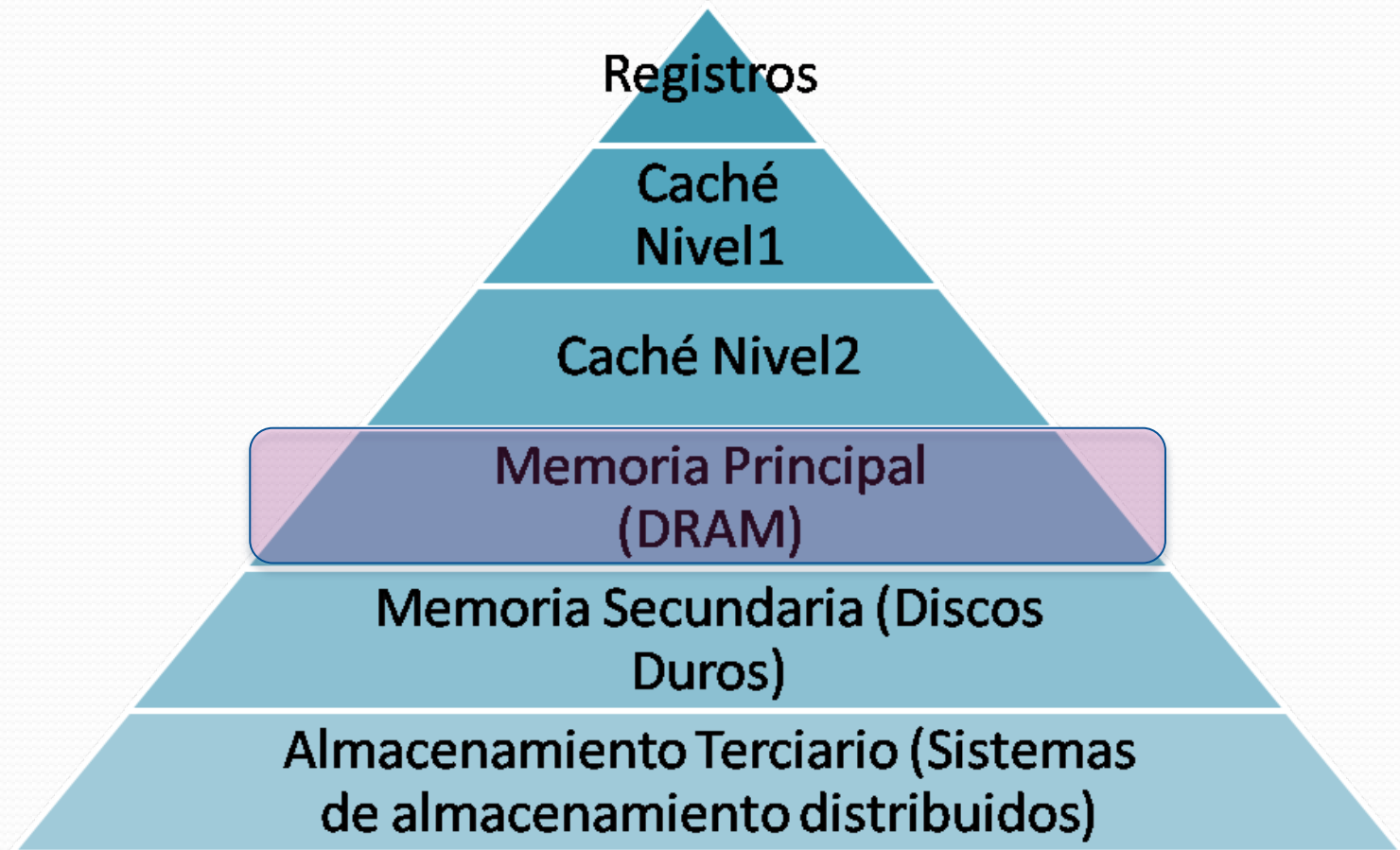


FIFO

9. GESTIÓN DE MEMORIA

- Asignar memoria principal a los procesos que la soliciten.
- Controlar zonas de memoria libres y ocupadas.
- Evitar que un proceso acceda a una zona de memoria de otro proceso.
- Gestionar la memoria compartida: algunos procesos pueden necesitar compartir información → acceden a la misma sección de memoria.

9. GESTIÓN DE MEMORIA



9. GESTIÓN DE MEMORIA

- Gestionar el intercambio entre memoria principal y memoria secundaria. Se produce un intercambio entre la memoria principal y el disco cuando la memoria principal no tiene capacidad para todos los procesos que tienen necesidad de ella.
- Memoria virtual: permite simular una RAM de mayor tamaño.

9. GESTIÓN DE MEMORIA

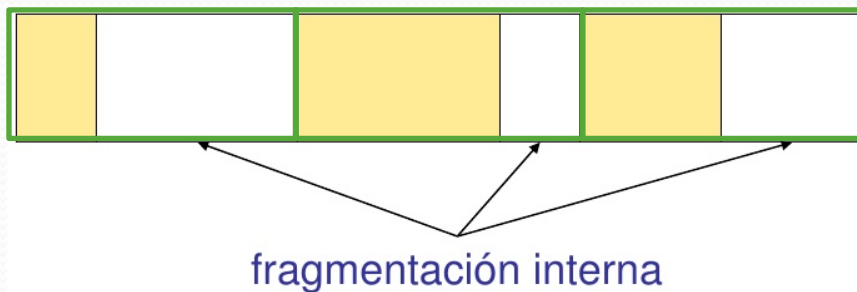
Funcionamiento

- Para que un programa se ejecute su código y sus datos deben estar cargados en memoria.
- En un sistema multitarea la memoria debe repartirse entre los distintos procesos.
- Algunos de esos procesos corresponden al S.O.
- Los retos de la gestión de memoria son: dar servicio a todos los procesos, evitar la fragmentación y no perjudicar el rendimiento (tiempos de acceso)

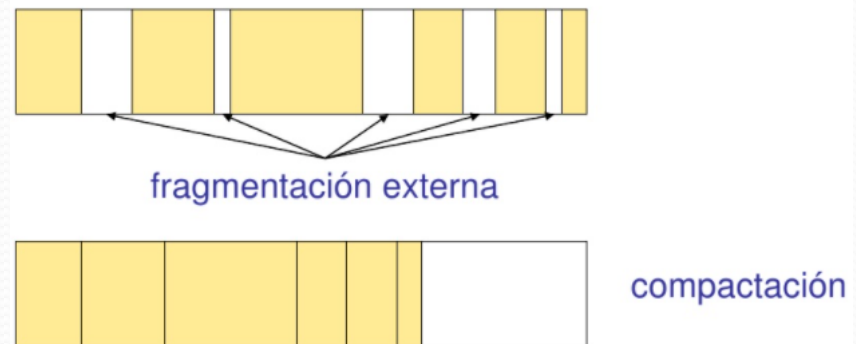
9. GESTIÓN DE MEMORIA

Técnicas utilizadas por el administrador de memoria

- Partición fija: se asignan franjas de memoria de tamaño fijo a cada proceso.
- Partición dinámica: se asigna a cada proceso el tamaño en memoria que necesite.



Partición fija

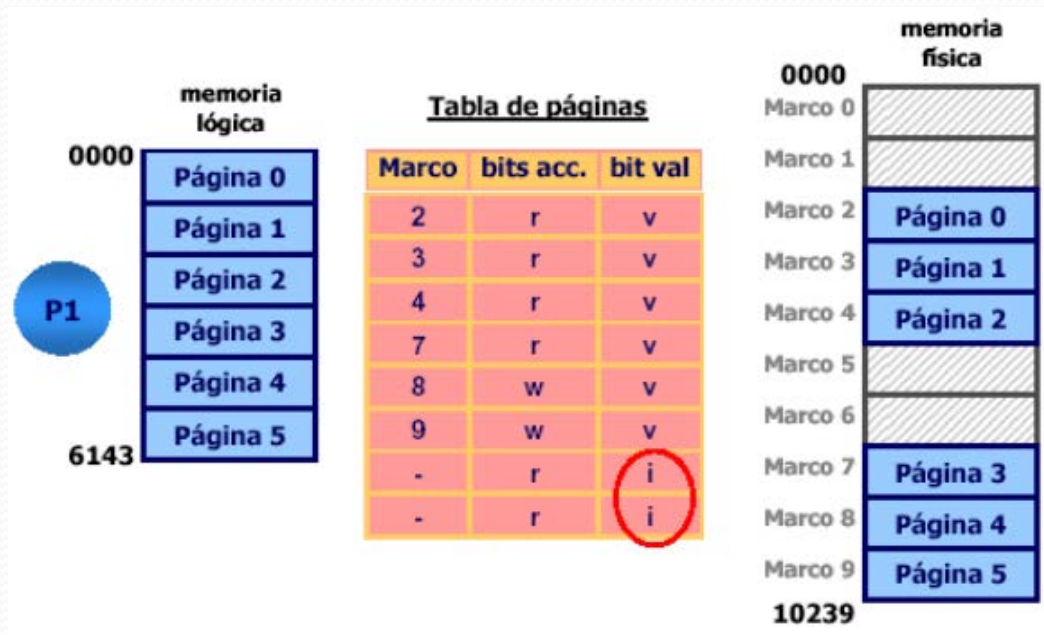


Partición variable

9. GESTIÓN DE MEMORIA

Técnicas utilizadas por el administrador de memoria

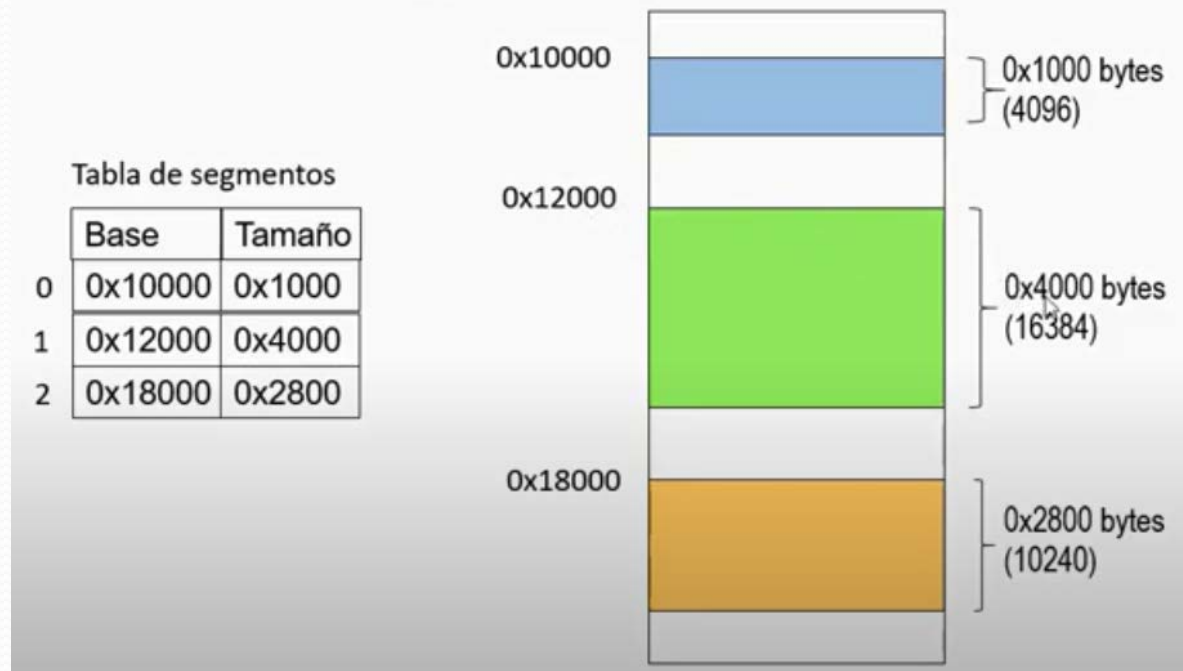
- Paginación: la memoria se divide en marcos y los procesos en páginas del tamaño de esos marcos.
Tabla de páginas.



9. GESTIÓN DE MEMORIA

Técnicas utilizadas por el administrador de memoria

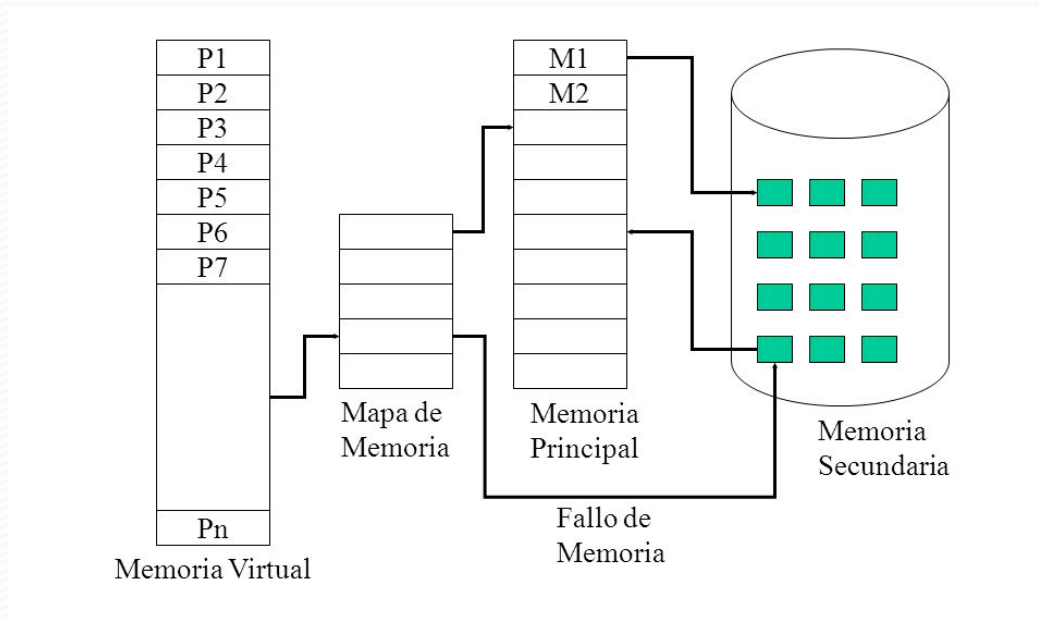
- Segmentación: cada proceso se divide en segmentos de longitud variable. Tabla de segmentos.



9. GESTIÓN DE MEMORIA

Técnicas utilizadas por el administrador de memoria

- Memoria virtual: técnica que permite al software utilizar más memoria principal de la que realmente existe



9. GESTIÓN DE MEMORIA

Memoria virtual

- Si los programas se pueden ejecutar por partes, la memoria lógica puede ser mayor que la real disponible.
- Al ejecutar un programa se carga solo la parte de instrucciones y datos necesarios en la memoria, dejando el resto en el disco.
- Cuando necesite las partes que están en disco las traerá a la memoria y devolverá al disco lo anterior que ya no es necesario.

10. GESTIÓN DE E/S

Una de las funciones del S.O. es procesar la información que proviene o que muestra en los periféricos.

Con el término E/S se designa cualquier transferencia de información desde el exterior y hacia la memoria o el procesador. También la transferencia en sentido contrario.

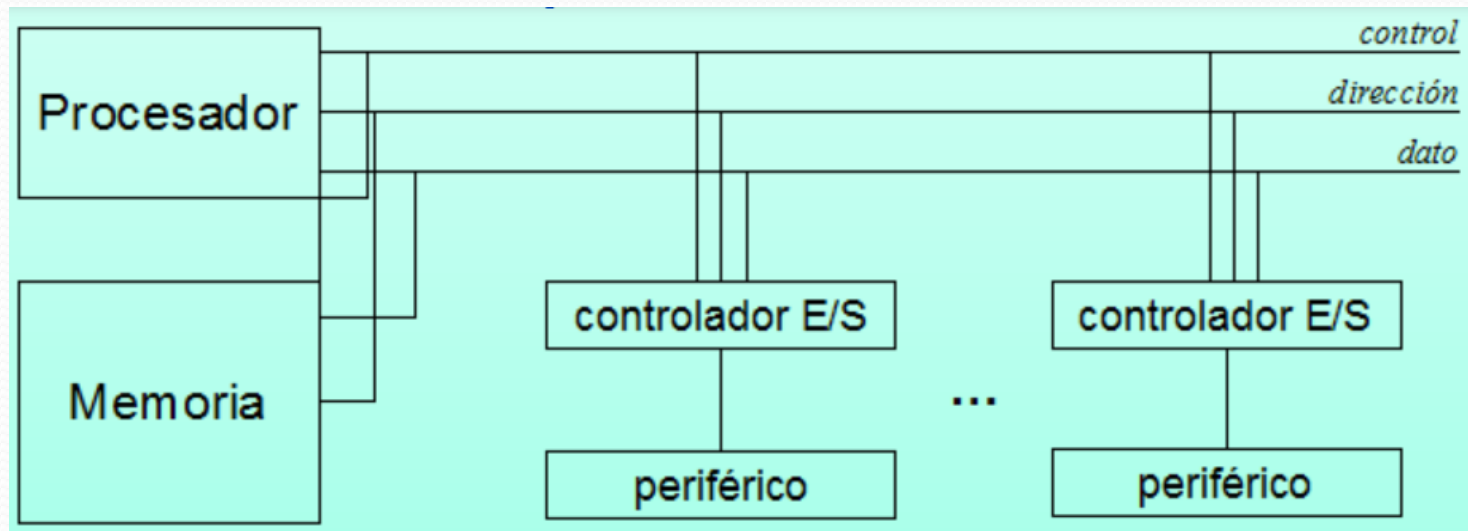
Un periférico tiene dos partes:

- Un controlador, que es el software que se encarga de la comunicación con la CPU
- Un dispositivo mecánico, electromecánico o electromagnético

10. GESTIÓN DE E/S

Controlador

Es un módulo que implementa la interfaz de los periféricos con el procesador y la memoria. Oculta los detalles físicos, de formato, temporización, ...



10. GESTIÓN DE E/S

Las operaciones de E/S pueden ser de varios tipos

- **E/S programada:** es la CPU la que se encarga de todo, por lo que dicho elemento puede estar sobrecargado.

El procesador envía una orden y espera a que la operación termine comprobando periódicamente el estado del controlador hasta que vea que la operación ha terminado.

10. GESTIÓN DE E/S

- **E/S por interrupciones:** la CPU ejecuta las transferencias cuando así lo requiere el periférico pero no está pendiente de forma continua.

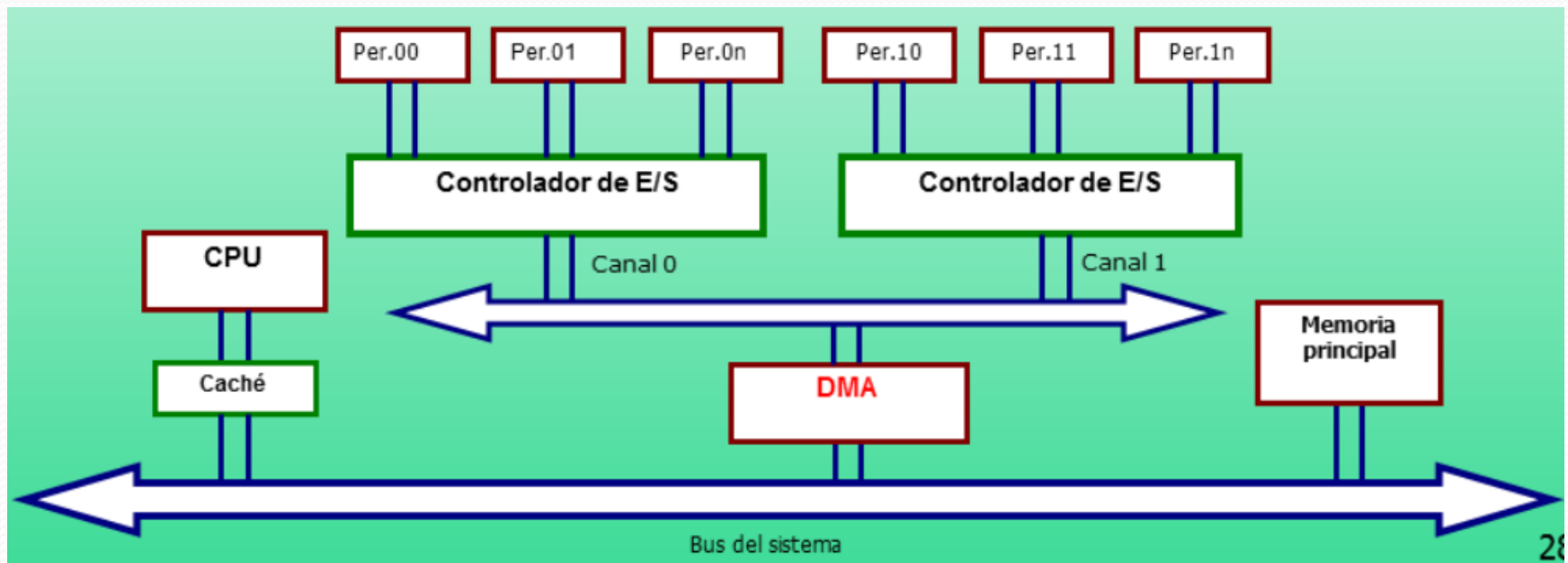
El procesador da la orden y sigue ejecutando otras instrucciones hasta que el controlador de E/S le interrumpe para comunicarle que la operación ha concluido.

10. GESTIÓN DE E/S

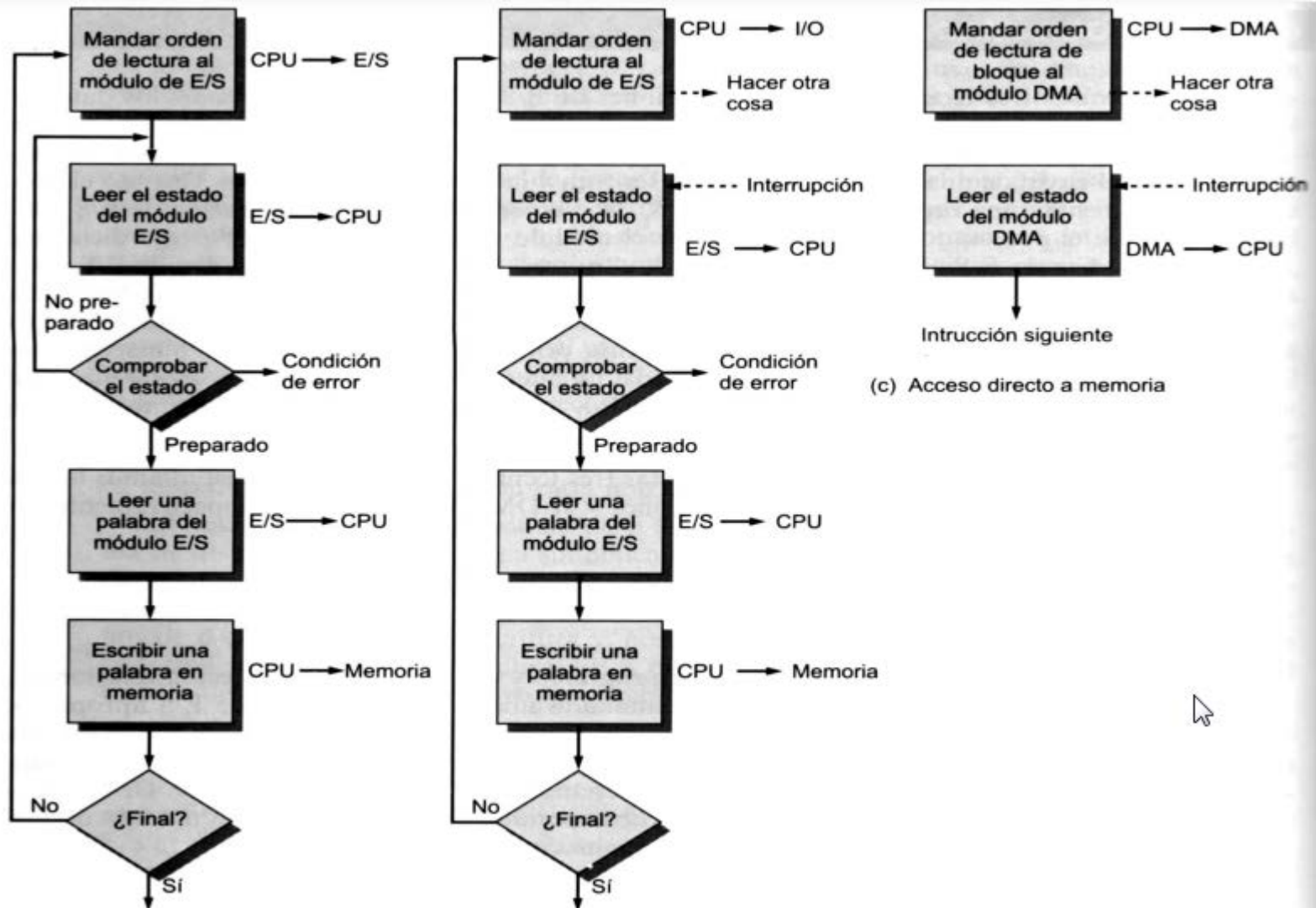
- **Acceso directo a memoria (DMA):** la memoria y el controlador de E/S intercambian datos directamente sin la intervención del procesador.

El procesador da la orden y sigue ejecutando otras instrucciones hasta que el controlador de E/S le interrumpe para comunicarle que la operación ha concluido, pero los datos no pasan a través de él.

10. GESTIÓN DE E/S



10. GESTIÓN DE E/S



11. GESTIÓN SISTEMA DE ARCHIVOS

Gestiona el almacenamiento de la información. Permite crear, modificar y borrar archivos y directorios.

Maneja el nivel lógico del sistema de archivos ocultando el nivel físico al usuario (pistas, sectores, cilindros, ...)

Archivos o ficheros → Estructuras de datos donde se almacenan la información y los programas. Estas estructuras dependen del sistema de archivos del S.O. y del tipo de fichero (extensión)

11. GESTIÓN SISTEMA DE ARCHIVOS

Características de los ficheros y directorios

- Nombre
- Extensión
- Permisos
- Propietario
- Fecha de creación
- Fecha del último acceso
- Fecha de la última modificación
- Tamaño

11. GESTIÓN SISTEMA DE ARCHIVOS

Operaciones con archivos

- Crear
- Borrar
- Abrir
- Cerrar
- Leer
- Escribir
- Renombrar
- Enlaces

Operaciones con directorios

- Crear
- Borrar
- Abrir
- Cerrar
- Leer
- Renombrar
- Enlaces

11. GESTIÓN SISTEMA DE ARCHIVOS

Rutas de acceso

La ruta de acceso a un archivo o directorio se indica nombrando todos los directorios y subdirectorios que tienen que atravesar hasta llegar al elemento concreto.

La forma de expresarlo depende del S.O.:

- Windows: se utiliza la barra \
C:\Windows\system32\drivers\acpi.sys
- Linux: se utiliza la barra /
/var/www/html/index.html

11. GESTIÓN SISTEMA DE ARCHIVOS

Rutas de acceso

- **Ruta de acceso absoluta**

Comienza desde el directorio raíz y va descendiendo en la estructura de directorios hasta llegar al archivo o directorio buscado.

- **Ruta de acceso relativa**

Se escribe la ruta a partir del directorio activo, lo que se indica con punto (.)

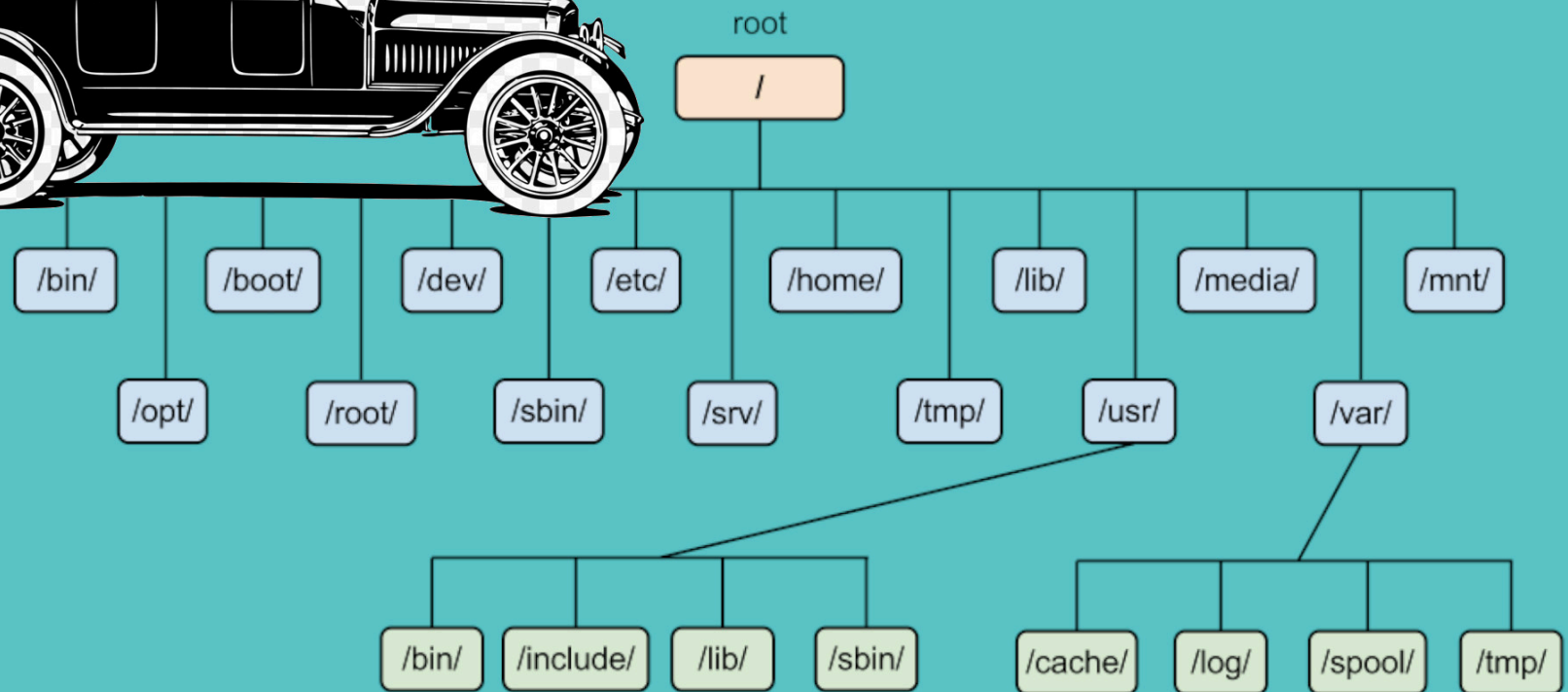
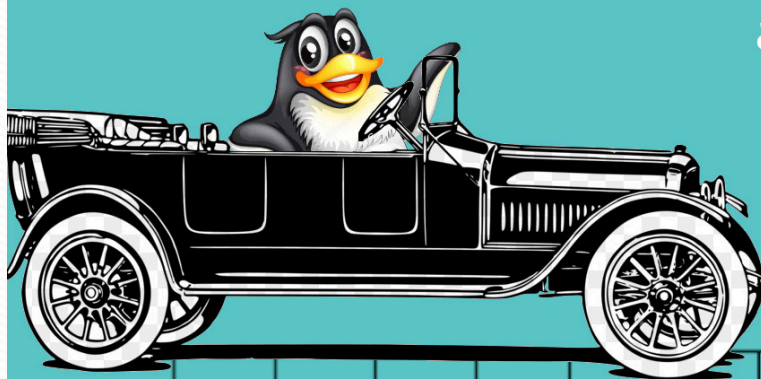
Al elemento padre o directorio de nivel superior se le referencia con dos puntos seguidos (..)

11. GESTIÓN SISTEMA DE ARCHIVOS

ABSOLUTA: Se indica toda la ruta, incluyendo el directorio raíz.
Ej: `ls /usr/bin/wireshark`

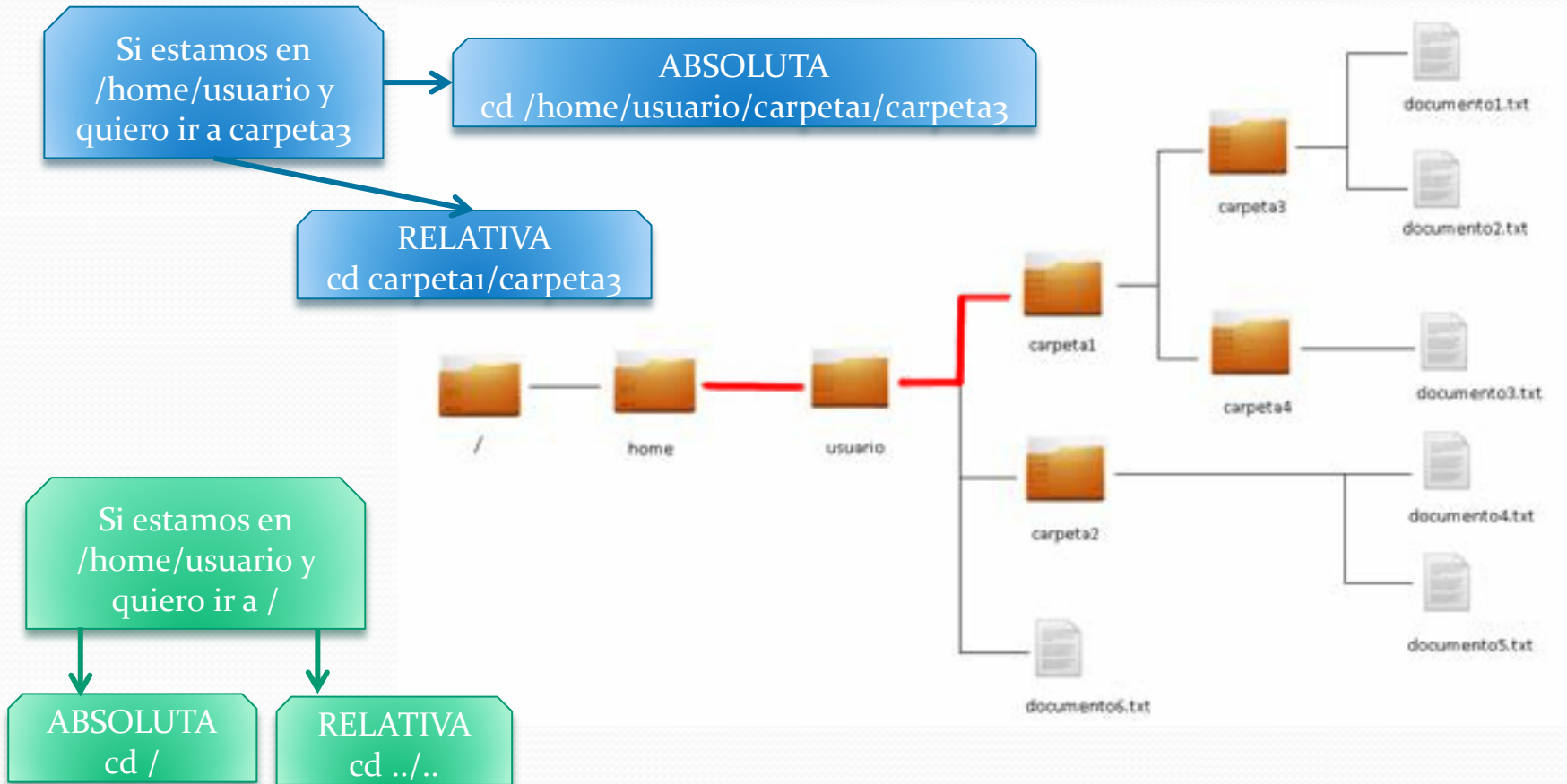
RELATIVA: Se indica la ruta a partir de donde esté situado.
Ej (desde cache): `ls ../spool`

Rutas relativas y absolutas



11. GESTIÓN SISTEMA DE ARCHIVOS

Rutas de acceso



12. GESTIÓN DE LA RED

Tareas fundamentales

- Gestión de los drivers de las tarjetas de red
- Gestión de los protocolos de comunicación
- Gestión de las aplicaciones en red
 - Programas de aplicación
 - Acceso a recursos remotos

13. PROTECCIÓN Y SEGURIDAD

Requisitos de seguridad de un S.O.

- **Confidencialidad:** los elementos del sistema solo serán visibles a los usuarios o grupos autorizados.
- **Integridad:** los elementos del sistema sólo serán modificados por los usuarios o grupos autorizados.
- **Disponibilidad:** los elementos del sistema estarán disponibles solo para usuarios o grupos autorizados.
- **Autenticación:** dar acceso al sistema solo a los usuarios o grupos autorizados.

13. PROTECCIÓN Y SEGURIDAD

Mecanismos de seguridad de un S.O.

- Control de acceso
- Uso limitado del usuario administrador / root
- Uso de contraseñas seguras
- Actualizaciones
- Antivirus
- Copias de seguridad
- Sistemas RAID
- Archivos LOG

14. GESTORES DE ARRANQUE

En un mismo ordenador se pueden instalar varios sistemas operativos.

Gestor de arranque: programa que permite seleccionar un sistema operativo en caso de arranque múltiple (varios sistemas operativos).

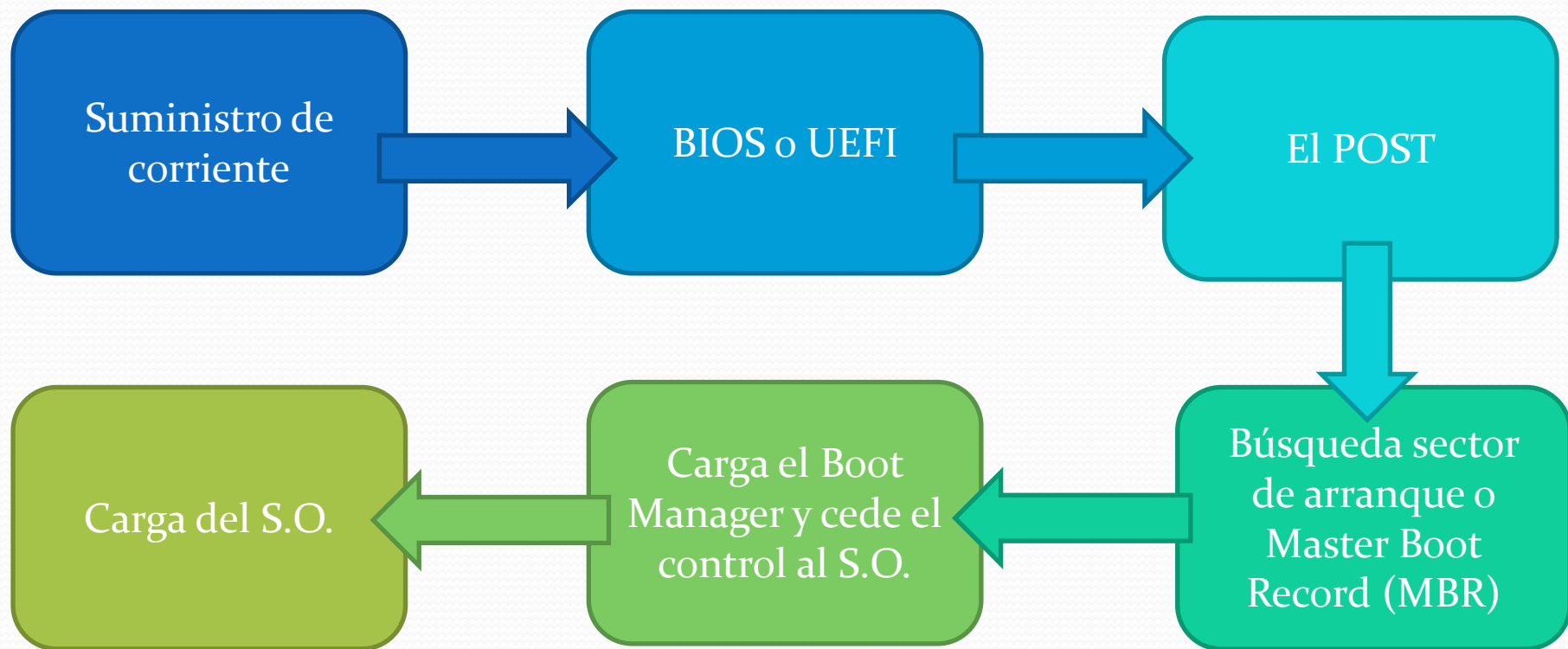
Se ejecuta después del inicio normal de la BIOS.

14. GESTORES DE ARRANQUE

- NTLDR (NT Loader): utilizado en el arranque de algunos sistemas Windows hasta Windows XP / Windows 2003 Server
- Bootmgr: utilizado también en sistemas Windows desde Windows Vista / Windows 2008 Server
- Lilo (Linux Loader)
- Grub: similar a Lilo pero más moderno y flexible

14. GESTORES DE ARRANQUE

Secuencia de arranque



15. TENDENCIAS ACTUALES

- Paralelismo: incremento de procesadores
- Sistemas distribuidos: incremento del uso de las redes
- Sistemas tolerantes a fallos
- Interfaces de usuario más amigables: interfaces gráficas, elementos multimedia, reconocimiento de voz, realidad virtual
- Sistemas abiertos: estandarización para que productos de distintos fabricantes sean compatibles
- Sistemas orientados a objetos
- Manejo de múltiples sistemas operativos