

Tema 6

Fundamentos de sistemas operativos

6-dic-10

Contenidos

- **Definición**
 - Objetivos y funciones
 - Contexto
 - Arranque de un computador
- **Gestión de recursos:**
 - Gestión del procesador,
 - Gestión de la memoria
 - Gestión de E/S
 - Gestión de archivos
- **Sistemas operativos más conocidos.**
- **Seguridad**

Contenidos

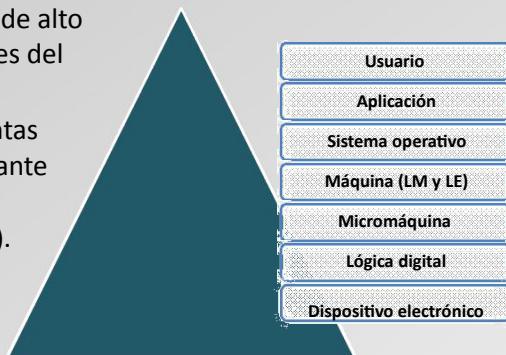
- **Definición** 
 - Objetivos y funciones
 - Contexto
 - Arranque de un computador
- **Gestión de recursos:**
 - Gestión del procesador,
 - Gestión de la memoria
 - Gestión de E/S
 - Gestión de archivos
- **Sistemas operativos más conocidos**
- **Seguridad**

Objetivos y funciones

- Un **Sistema Operativo** es un programa (o conjunto de programas) de **control** que tiene por objeto **facilitar** el uso del ordenador y conseguir que este se utilice **eficientemente**.
 - Funciones principales
 - Facilitar el uso del ordenador (comunicación ordenador/usuario)
 - Gestionar y asigna recursos Hw a los programas
 - Gestionar y mantener archivos (en la memoria externa)
 - Proteger los datos y los programas
 - Identificar y autenticar a los usuarios
 - Contabilizar la utilización de los recursos,
 - Etc.

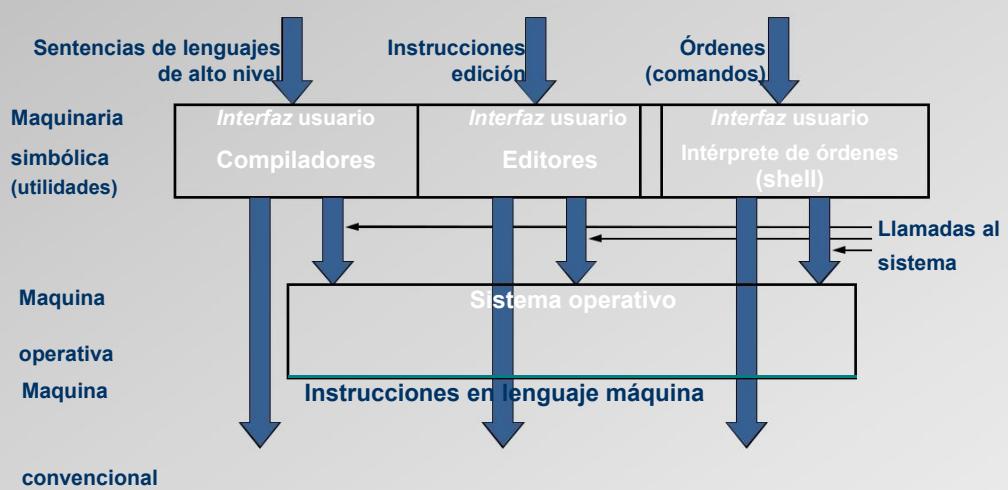
Contexto donde se encuadra el Sistema Operativo

- Los programas de aplicación y los usuarios “ven” al computador como una **máquina simbólica**, que proporciona lenguajes de alto nivel, editores y órdenes del SO.
- Este nivel utiliza las distintas utilidades del SO mediante **llamadas al sistema** (o instrucciones virtuales).



El nivel de máquina operativa

Usuarios y programas de aplicaciones



El nivel de máquina operativa

- Los usuarios y los programas se comunican con el SO mediante la **interfaz de usuario** que se basa en un **lenguaje de control o de órdenes (commands)** que es captado y traducido a llamadas al sistema por el intérprete de órdenes (**shell**).

Interfaz del SO

- **Los usuarios y los programas se comunican con el SO**
 - mediante la interfaz de usuario que se basa en un lenguaje de control o de órdenes que es captado y traducido a llamadas al sistema por el intérprete de órdenes (concha o shell).
- **La interfaz de usuario puede ser de:**
 - líneas de texto (líneas de órdenes), o
 - menús, o
 - iconos,
 - Ventanas, etc.



Interfaz SO: línea de órdenes

- Todos los programas / Accesorios / Símbolo del sistema



Ejemplos de órdenes del SO

- Date y Dir

```
C:\Users\Alberto>date
La fecha actual es: 05/01/2011
Escriba la nueva fecha: <dd-mm-aa>

C:\Users\Alberto>Dir
El volumen de la unidad C no tiene etiqueta.
El n mero de serie del volumen es: D8C8-0AE5

Directorio de C:\Users\Alberto

03/08/2010 22:37    <DIR>          Contacts
13/12/2010 11:39    <DIR>          Desktop
22/12/2010 20:18    <DIR>          Documents
09/12/2010 12:52    <DIR>          Downloads
04/11/2010 15:52    <DIR>          Favorites
03/08/2010 22:37    <DIR>          Links
03/08/2010 22:37    <DIR>          Music
05/01/2011 09:13    8.650.752 ntuser.dat
16/11/2010 13:12    <DIR>          Pictures
03/08/2010 22:37    <DIR>          Saved Games
03/08/2010 22:37    <DIR>          Searches
27/09/2010 17:51    <DIR>          Tracing
01/01/2011 13:18    <DIR>          Videos
                           1 archivos      8.650.752 bytes
                           12 dirs   168.540.151.808 bytes libres

C:\Users\Alberto>
```

Lista de órdenes (MS-DOS)

```

C:\Users\Alberto>date
La fecha actual es: 05/01/2011
Escribir la nueva fecha (dd-mm-aa)

C:\Users\Alberto>help
Para obtener más información acerca de un comando específico, escriba HELP
seguido del nombre del comando.

DIR Muestra una lista de archivos y subdirectorios en un
      directorio.

DISKCOMP Copia el contenido de un disco en otro.

DISKCOPY Copia el contenido de un disco en otra partición de disco.

DISKPART Administración de particiones de Windows y
      creación de discos.

DOSEXE Edita líneas de comando, memoriza comandos de Windows y
      crea macros.

DRIVERQUERY Muestra el estado y las propiedades actuales del controlador
      de dispositivo.

ECO Muestra mensajes, o activa y desactiva el eco.

EMBLLOC Muestra la búsqueda de variables del entorno del archivo por
      lotes.

ERASE Elimina los datos de un disco.

EXIT Sale del programa CMD.EXE (interfaz de comandos).

FC Compara dos archivos o conjunto de archivos y muestra las
      diferencias entre ellos.

FIND Busca una cadena de texto en uno o más archivos.

FINDSTR Busca cadenas de texto en uno o más archivos.

FOR Ejecuta un comando para cada archivo en un conjunto de
      archivos.

FORMAT Formatea un disco para usarlo con Windows.

FORMATIL Formatea un disco para usarlo con un conjunto de tipos de archivo.

FTYPE Muestra o modifica los tipos de archivo usados en una
      asociación de extensión de archivo.

GOTO Directiva que ejecuta una parte de las instrucciones de Windows a una línea
      en un programa por lotes.

GPRESULT Muestra las licencias y privilegios de grupo por equipo o usuario.

GPRINT Permite a Windows mostrar un Juego de Caracteres extendidos
      en modo gráfico.

HELP Muestra la descripción de ayuda para los comandos de Windows.

I2CCLS Muestra, modifica, hace copias de seguridad y restaura listas
      de control de acceso (ACL) para archivos y directorios.

IF Ejecuta una instrucción condicionalmente dependiendo de un valor.

LABEL Crea, cambia o elimina la etiqueta del volumen de un disco.

MD Crea un directorio.

RD Elimina un directorio.

MKLINK Crea vínculos simbólicos y vínculos físicos

MORE Muestra la información por pantalla.

POUE Muestra uno o más archivos de un directorio a otro en la
      misma unidad.

PROMPT Muestra o establece una ruta de búsqueda para archivos
      ejecutables.

RBU Inicia una copia de seguridad del sistema y después lo cambia.

RECOVER Recupera la información legible de un disco dañado o
      defectuoso.

REN Renombra archivos.

RENAME Cambia el nombre de uno o más archivos.

REPLACE Cambia el nombre de uno o más archivos.

REPLACELM Reemplaza archivos.

RDML Utilidad para mover o copiar árboles de archivos y directorios.

SET Muestra, establece o quita variables de entorno de Windows.

SETLOCAL Muestra, establece o quita variables de entorno en un
      archivo por lotes.

SC Muestra o configura servicios (procesos en segundo plano).

SERVICES Muestra comandos y programas para ejecutarlos en un equipo.

SHIFT SHIFTDIR Cambia posiciones de modificadores reemplazables en archivos
      por lotes.

SHUTDOWN Puede iniciar el apagado local o remoto de un equipo.

SORT Ordena la salida.

START Muestra otra ventana para ejecutar un programa o comando.

SUBST Muestra las propiedades y la configuración específicas
      del equipo.

TASKLIST Muestra las tareas en ejecución, incluidos los servicios.

TASKKILL Termina o interrumpe un proceso o aplicación que se está
      ejecutando.

TIME Muestra y establece la hora del sistema.

TITLE Establece el título de la ventana de una sesión de CMD.EXE.

TREE Muestra gráficamente la estructura de directorios de una
      unidad o ruta de acceso.

DIR DIR Muestra una lista de archivos y subdirectorios en un
      directorio.

DISKCOMP DISKCOMP Copia el contenido de un disco en otro.

DISKCOPY DISKCOPY Copia el contenido de un disco en otra partición de disco.

DISKPART DISKPART Administración de particiones de Windows y
      creación de discos.

DOSEXE DOSEXE Edita líneas de comando, memoriza comandos de Windows y
      crea macros.

DRIVERQUERY DRIVERQUERY Muestra el estado y las propiedades actuales del controlador
      de dispositivo.

ECO ECO Muestra mensajes, o activa y desactiva el eco.

EMBLLOC EMBLLOC Muestra la búsqueda de variables del entorno del archivo por
      lotes.

ERASE ERASE Elimina los datos de un disco.

EXIT EXIT Sale del programa CMD.EXE (interfaz de comandos).

FC FC Compara dos archivos o conjunto de archivos y muestra las
      diferencias entre ellos.

FIND FIND Busca una cadena de texto en uno o más archivos.

FINDSTR FINDSTR Busca cadenas de texto en uno o más archivos.

FOR FOR Ejecuta un comando para cada archivo en un conjunto de
      archivos.

FORMAT FORMAT Formatea un disco para usarlo con Windows.

FORMATIL FORMATIL Formatea un disco para usarlo con un conjunto de tipos de archivo.

FTYPE FTYPE Muestra o modifica los tipos de archivo usados en una
      asociación de extensión de archivo.

GOTO GOTO Directiva que ejecuta una parte de las instrucciones de Windows a una línea
      en un programa por lotes.

GPRESULT GPRESULT Muestra las licencias y privilegios de grupo por equipo o usuario.

GPRINT GPRINT Permite a Windows mostrar un Juego de Caracteres extendidos
      en modo gráfico.

HELP HELP Muestra la descripción de ayuda para los comandos de Windows.

I2CCLS I2CCLS Muestra, modifica, hace copias de seguridad y restaura listas
      de control de acceso (ACL) para archivos y directorios.

IF IF Ejecuta una instrucción condicionalmente dependiendo de un valor.

LABEL LABEL Crea, cambia o elimina la etiqueta del volumen de un disco.

MD MD Crea un directorio.

RD RD Elimina un directorio.

MKLINK MKLINK Crea vínculos simbólicos y vínculos físicos

MORE MORE Muestra la información por pantalla.

POUE POUE Muestra uno o más archivos de un directorio a otro en la
      misma unidad.
  
```

Lista de órdenes (MS-DOS)

```

MOVE Mueve uno o más archivos de un directorio a otro en la
      misma unidad.

OPENFILES Muestra los archivos compartidos abiertos por usuarios remotos.

PATH Muestra o establece una ruta de búsqueda para archivos
      ejecutables.

PRUNE Interrumpe el proceso de un archivo por lotes y muestra un
      mensaje.

POPD Restaura el valor anterior del directorio actual guardado
      por PUSHD.

PRINT Imprime un archivo de texto.

PROMPT Cambia el símbolo de comandos de Windows.

RBU Cambia el nombre de uno o más archivos.

RD Quitar un directorio.

RECOVER Recupera la información legible de un disco dañado o
      defectuoso.

REM Marca comentarios en archivos por lotes o CONFIG.SYS.

REN Cambia el nombre de uno o más archivos.

RENAME Cambia el nombre de uno o más archivos.

REPLACE Reemplaza archivos.

RDML Utilidad para mover o copiar árboles de archivos y directorios.

SET Muestra, establece o quita variables de entorno de Windows.

SETLOCAL Muestra, establece o quita variables de entorno en un
      archivo por lotes.

SC Muestra o configura servicios (procesos en segundo plano).

SERVICES Muestra comandos y programas para ejecutarlos en un equipo.

SHIFT SHIFTDIR Cambia posiciones de modificadores reemplazables en archivos
      por lotes.

SHUTDOWN Puede iniciar el apagado local o remoto de un equipo.

SORT Ordena la salida.

START Muestra otra ventana para ejecutar un programa o comando.

SUBST Muestra las propiedades y la configuración específicas
      del equipo.

TASKLIST Muestra las tareas en ejecución, incluidos los servicios.

TASKKILL Termina o interrumpe un proceso o aplicación que se está
      ejecutando.

TIME Muestra y establece la hora del sistema.

TITLE Establece el título de la ventana de una sesión de CMD.EXE.

TREE Muestra gráficamente la estructura de directorios de una
      unidad o ruta de acceso.

TYPE TYPE Muestra el contenido de un archivo de texto.

VER Muestra la versión de Windows.

VERIFY Comunica a Windows si debe comprobar que los archivos se
      escriben de forma correcta en un disco.

VOL Muestra la etiqueta del volumen y el número de serie del disco.

XCOPY XCOPY Copia archivos y árboles de directorios.

WMIC WMIC Muestra información de WMI en el shell de comandos
      interactivo.

DIR DIR Muestra una lista de archivos y subdirectorios en un
      directorio.

DISKCOMP DISKCOMP Copia el contenido de un disco en otro.

DISKCOPY DISKCOPY Copia el contenido de un disco en otra partición de disco.

DISKPART DISKPART Administración de particiones de Windows y
      creación de discos.

DOSEXE DOSEXE Edita líneas de comando, memoriza comandos de Windows y
      crea macros.

DRIVERQUERY DRIVERQUERY Muestra el estado y las propiedades actuales del controlador
      de dispositivo.

ECO ECO Muestra mensajes, o activa y desactiva el eco.

EMBLLOC EMBLLOC Muestra la búsqueda de variables del entorno del archivo por
      lotes.

ERASE ERASE Elimina los datos de un disco.

EXIT EXIT Sale del programa CMD.EXE (interfaz de comandos).

FC FC Compara dos archivos o conjunto de archivos y muestra las
      diferencias entre ellos.

FIND FIND Busca una cadena de texto en uno o más archivos.

FINDSTR FINDSTR Busca cadenas de texto en uno o más archivos.

FOR FOR Ejecuta un comando para cada archivo en un conjunto de
      archivos.

FORMAT FORMAT Formatea un disco para usarlo con Windows.

FORMATIL FORMATIL Formatea un disco para usarlo con un conjunto de tipos de archivo.

FTYPE FTYPE Muestra o modifica los tipos de archivo usados en una
      asociación de extensión de archivo.

GOTO GOTO Directiva que ejecuta una parte de las instrucciones de Windows a una línea
      en un programa por lotes.

GPRESULT GPRESULT Muestra las licencias y privilegios de grupo por equipo o usuario.

GPRINT GPRINT Permite a Windows mostrar un Juego de Caracteres extendidos
      en modo gráfico.

HELP HELP Muestra la descripción de ayuda para los comandos de Windows.

I2CCLS I2CCLS Muestra, modifica, hace copias de seguridad y restaura listas
      de control de acceso (ACL) para archivos y directorios.

IF IF Ejecuta una instrucción condicionalmente dependiendo de un valor.

LABEL LABEL Crea, cambia o elimina la etiqueta del volumen de un disco.

MD MD Crea un directorio.

RD RD Elimina un directorio.

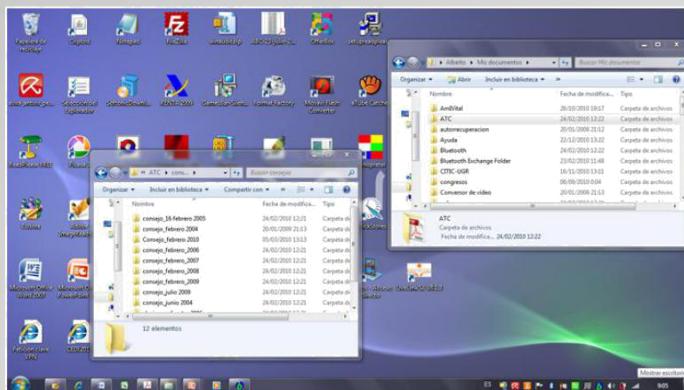
MKLINK MKLINK Crea vínculos simbólicos y vínculos físicos

MORE MORE Muestra la información por pantalla.

POUE POUE Muestra uno o más archivos de un directorio a otro en la
      misma unidad.
  
```

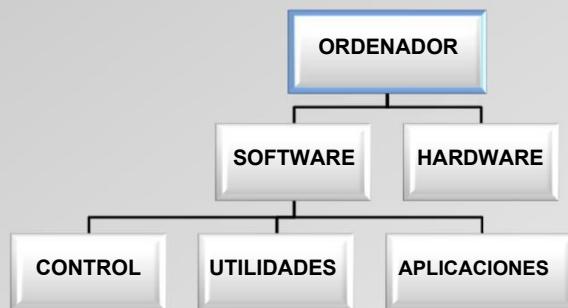
Interfaz del SO: GUI

- Para los usuarios la forma más cómoda de utilizar un programa (y el SO, en concreto) es mediante una **GUI (Interfaz Gráfica de Usuario)**. Por ejemplo, la interfaz Windows.



El SO en el contexto del software del computador

- **Soporte lógico o software de un computador**
 - es el conjunto de programas asociados a dicho computador: los suministrados por el constructor, los adquiridos en empresas especializadas, y los redactados por los propios usuarios.



Software de control

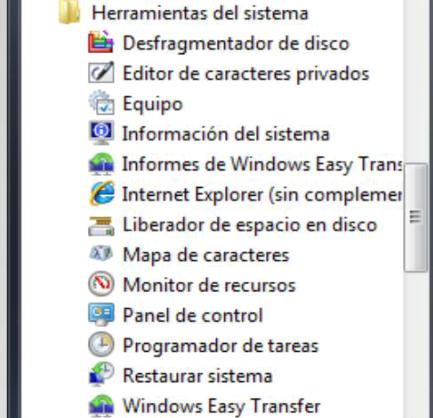
- Módulos que controlan el funcionamiento de los programas que se ejecutan, y administran los recursos hardware, facilitando el uso del computador de la forma más eficiente posible.
- Dentro de este apartado se incluye:
 - El SO
 - Intérprete de órdenes
 - Programas de diagnóstico y mantenimiento

Utilidades

- Programas genéricos de servicio que son como una ampliación del SO.
 - Por ejemplo: programas para compresión de datos, copias de seguridad, compactación de discos, recuperación de archivos, antivirus, respaldo de seguridad, compresión de datos, etc.
- Herramientas generales que ayudan al usuario en la construcción de aplicaciones.
 - Por ejemplo: compiladores, intérpretes, cargadores montadores de programas, editores de textos, rastreadores-depuradores, administrador de bibliotecas de programas, etc.

Ejemplo de utilidades (Windows 7)

- Inicio ➔ Todos los programas ➔ Accesorios ➔ Herramientas del sistema



Software de aplicaciones

- incluye programas relacionados con aplicaciones específicas tanto adquiridos como realizados por los propios usuarios:
- bibliotecas de programas matemáticos y estadísticos
- Ofimática: procesador de textos, programa de presentaciones, programa de hoja electrónicas.
- Navegador de web
- Gestos de usuario de correo electrónico
- sistemas de administración de archivos y de bases de datos
- programas para gestión de comunicaciones (correo electrónico, etc.)
- visualizadores y navegadores de Web
- programas de aplicaciones gráficas
- programas CAD/CAM
- videojuegos,
- programas de los usuarios

Arranque del ordenador (*booting*)

- Al encender el ordenador:

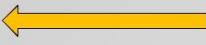
- Programa de autodiagnóstico de encendido (ROM): identifica memoria, discos, teclado, ratón,..
- Cargador inicial (ROM)

AE Residente

- Invitación a que el usuario inicie su trabajo: intérprete de órdenes / escritorio



Contenidos

- **Definición**
 - Objetivos y funciones
 - Contexto
 - Arranque de un computador
- **Gestión de recursos:** 
 - Gestión del procesador,
 - Gestión de la memoria
 - Gestión de E/S
 - Gestión de archivos
- **Sistemas operativos más conocidos**
- **Seguridad**

Gestión de recursos

Sistema operativo

Gestión del procesador

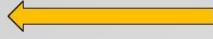
Gestión de memoria

Gestión de E/S

Gestión de archivos

Contenidos

- **Definición**
 - Objetivos y funciones
 - Contexto
 - Arranque de un computador
- **Gestión de recursos:**
 - Gestión del procesador
 - Gestión de la memoria
 - Gestión de E/S
 - Gestión de archivos
- **Sistemas operativos más conocidos**
- **Seguridad**



Gestión del procesador

- **Aspectos a considerar:**

- Monoprogramación y multiprogramación
- Intercambio entre MP y disco
- Estados de un proceso

Gestión del procesador

- **Proceso**

- es un programa que ha iniciado su ejecución.

- **El proceso**

- nace en el momento que se inicia su ejecución y muere en el momento que finaliza su ejecución o que se cancela por algún motivo.

- **El SO gestiona la ejecución de procesos**

- monoprogramación
- multiprogramación

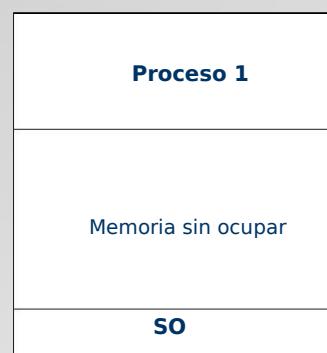
Gestión del procesador

- **Tradicionalmente los programas a ejecutar:**
 - se introducían en cinta o disco formando una cola (lote de trabajos, “batch”) en donde cada uno de ellos esperaba a que le tocase su turno de ejecución.
- **El SO:**
 - selecciona del lote de trabajos pendientes los programas a ejecutar en cada momento
 - Esto se denomina **procesamiento por cola serie** o “**procesamiento por lotes**” (“batch”)
- **La selección del programa a ejecutar**
 - se realiza, de acuerdo con un esquema de prioridades, por un módulo de SO denominado **planificador de trabajos**, o **planificador a largo plazo**

Monoprogramación

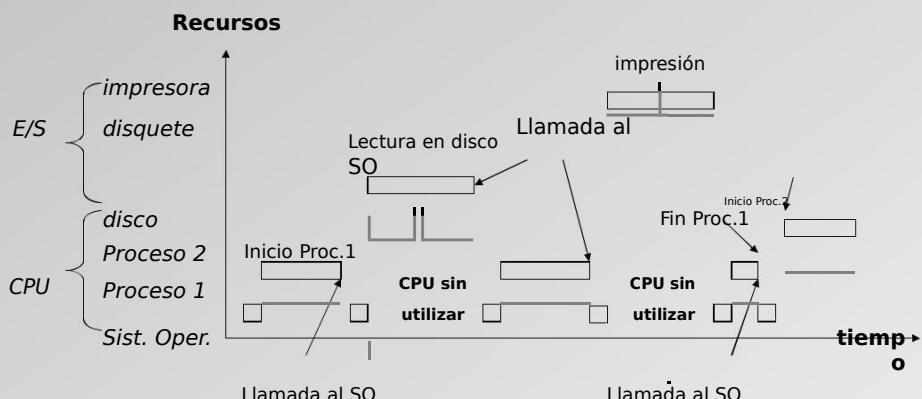
- En la monoprogramación se ubica en la Memoria Principal un programa (además del SO).
- No se inicia la ejecución de otro programa hasta que finaliza totalmente el que se encuentra en ejecución.
- Concepto de procesamiento “por lotes”, o cola serie (“batch”)

Memoria principal



Monoprogramación

- Hay un solo programa en MP y hasta que no termina su ejecución no comienza otro.



Funciones del SO en la gestión de monoprogramación

- Siempre que un programa tiene una **operación de E/S** hace una **llamada al SO**, encargándose éste de generar las instrucciones máquina concretas de la E/S.
- Al **finalizar una operación de E/S** se genera una **interrupción** que provoca una **llamada al SO**, y continua la ejecución por la CPU del proceso. Un proceso en ejecución se dice que está **activo**.
- Mientras se efectúa la operación de E/S el proceso queda **bloqueado**, ya que la CPU no puede continuar la ejecución del proceso hasta que acabe la operación de E/S.
- Al acabar la ejecución de un proceso, el planificador a largo plazo del SO, selecciona un programa de la cola serie, lo **carga en la MP** e **inicia su ejecución**.

Inconvenientes de la monoprogramación

- **Consecuencias:**

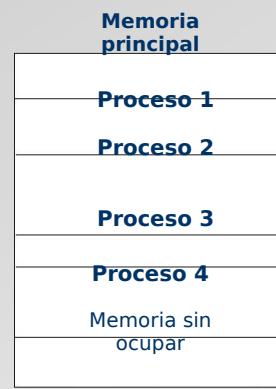
- La memoria casi siempre queda parcialmente ocupada (cabrían más procesos en ella)
 - se desaprovecha la MP
- Siempre que hay una operación de E/S, la CPU no se utiliza. Estos tiempos suelen ser mucho mayores (7 u 8 órdenes de magnitud) que los tiempos de la CPU
 - se desaprovecha el procesador
- Los periféricos se utilizan uno a uno
 - se desaprovechan los periféricos

- **Conclusión**

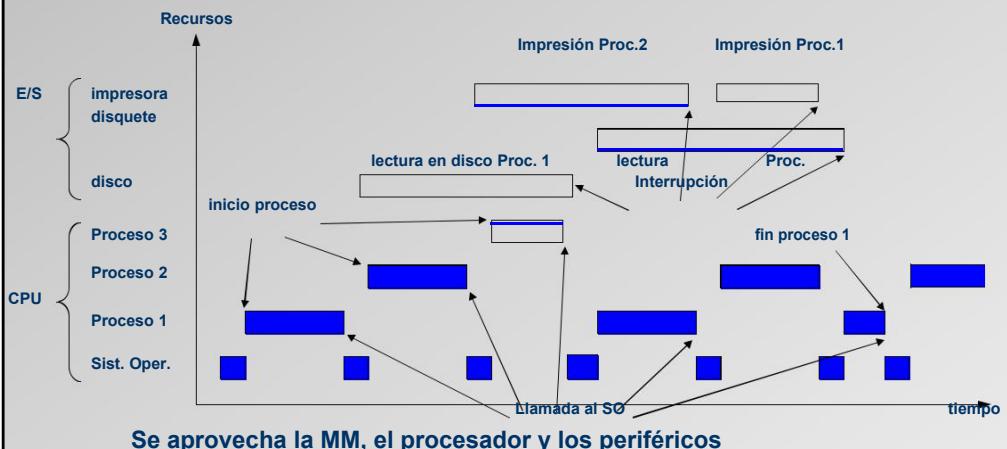
LA MONOPROGRAMACIÓN NO ES EFICIENTE

Multiprogramación

- **El SO carga en la MP todos los programas de la cola que quepan en ella**
- **No queda ningún programa en la cola que quepa en la partición libre de memoria**
- **El planificador de trabajos a corto plazo del SO asigna el procesador sucesivamente a los procesos preparados de forma que éste se aproveche al máximo.**
- **Los distintos procesos van avanzando en su ejecución.**



Procesamiento concurrente (multiprogramación clásica)



MULTIprogramación frente a MONOprogramación

- No hay que esperar a que acabe un proceso para cargar otro.
- Se aprovechan los tiempos muertos del procesador que se tenían con la monoprogramación, los tiempos muertos en periféricos y el espacio no ocupado en la MP.
- Todos los procesos van avanzando poco a poco. Se dice que se ejecutan “concurrentemente”
- La ejecución de los procesos se va solapando en el tiempo. Al usuario le parece que todo avance simultáneamente (cosa que no es posible ya que sólo hay un procesador).

Evolución de los sistemas de multiprogramación

- **En la multiprogramación clásica:**

- Se pasa a ejecutar otro proceso cuando se bloquea el que este en ejecución (se dice que el sistema operativo es “**no apropiativo**”)
- Puede haber procesos “usureros”

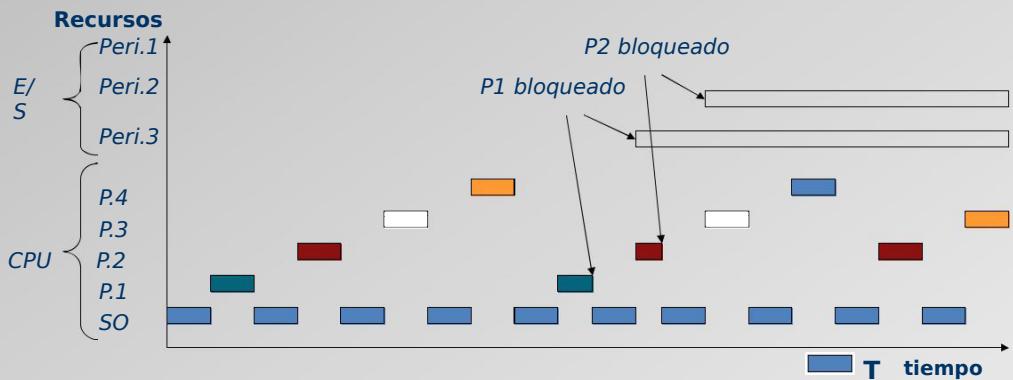
- **Multiprogramación más evolucionada:**

- El SO puede interrumpir la ejecución de un proceso:
 - Cuando llegue a la cola un programa de MAYOR preferencia.
 - Cuando un proceso quede bloqueado, por estar efectuándose una operación de E/S (Interrupción de E/S).
 - Etc.

Sistema operativo de tiempo compartido ("Time sharing")

- A cada proceso se le asigna un quantum de tiempo ($T=20\text{ms}$, p.e.) al final del cuál se interrumpe su ejecución. La interrupción lanza el módulo **planificador a corto plazo (despachador)**, que selecciona para ejecutar a uno de los procesos preparados (no bloqueados) residentes en memoria.
- Se aprovechan al máximo todos los recursos y se da mayor sensación de que todos los procesos avanzan en su ejecución “simultáneamente” (en realidad “concurrentemente”).

Gestión de tiempo compartido (“turno rotatorio”)



Gestión de la multiprogramación

- **Bloque de control de procesos (BCP)**
 - El SO para planificar la ejecución de los procesos, mantiene por cada uno de ellos en la MP y en disco un BCP, que contiene información tal como: identificadores (del proceso y del usuario), contenidos de la CPU, y estado/privilegios/prioridad del proceso.
- **Cambio de contexto**
 - Para interrumpir (suspender) la ejecución de un proceso, con objeto de pasar a ejecutar otro, el SO debe efectuar un cambio de contexto, consistente en:
 - Salvar los contenidos de los registros del procesador, incluyendo palabra de estado, puntero de pila y contador de programa.
 - Restaurar en el procesador los valores anteriores correspondientes al proceso que va a continuar su ejecución.

Estados de un proceso (resumen) en multiprogramación



Concepto de hebra (*thread*)

- El SO UNIX introdujo el concepto de “hebra”, que incluyen los SO modernos:

– Un proceso puede descomponerse en tareas (hebra) diferentes e independientes, que se pueden separar y ejecutar en el tiempo concurrentemente (en paralelo) con otros procesos, o hebras, incluso del propio proceso (cada hebra tiene su estado, prioridad, privilegios, mapa de MV, etc.)

Ejemplos de hebras

- **Interfaz de órdenes del Windows:** Cada ventana de carpeta lleva asociado un hilo independiente de ejecución (pueden estar haciendo concurrentemente 2 copias de ficheros en ventanas diferentes).
- **Procesador de texto.** Cuando se lanza a ejecución (picando el ícono correspondiente) se genera un proceso que puede descomponerse en distintos hilos, cada uno de ellos con los distintos cometidos: leer las entradas del teclado (1er plano), visualiza de los menús, ejecutar las órdenes, corrector ortográfico, etc.

Medida de prestaciones

- **Productividad o rendimiento:**
 - es el número de trabajos ejecutados durante un periodo de tiempo preestablecido (12 horas, por ejemplo). Representa una medida del aprovechamiento del computador, en su totalidad.
 - Objetivo: Maximizar el uso del procesador
 - Ejemplo: Sistema que procesa 850 transacciones por segundo
 - **Tasa de utilización del procesador (RCPU)**
 - es, para un determinado intervalo de tiempo, el tiempo de uso del procesador con respecto al tiempo total:
 - Objetivo: Maximizar el uso del procesador
- $$\text{RCPU} = \frac{\text{tiempo de uso de procesador}}{\text{tiempo total}}$$

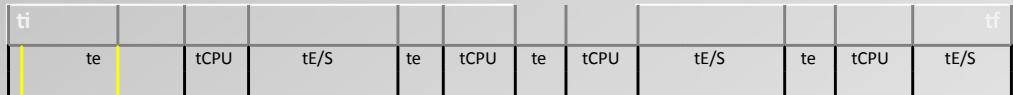
Medida de prestaciones

- **Tiempo de procesamiento o de CPU de un proceso (tCPU)**
 - Suma de los intervalos de tiempo en que el proceso está activo.
- **Tiempo de E/S de un proceso (tES)**
 - suma de intervalos de tiempos en que el proceso está bloqueado por operaciones de entrada/salida.
- **Tiempo de espera de un proceso (te):**
 - Tiempo que un proceso pasa en colas de espera (cola de entrada + preparado, esperando a su turno)

Medida de prestaciones

- **Tiempo de procesamiento o de CPU (tCPU)**
 - Suma de los intervalos de tiempo en que el proceso está activo.
- **Tiempo de E/S (tES)**
 - suma de intervalos de tiempos en que el proceso está bloqueado por operaciones de entrada/salida.
- **Tiempo de servicio (ts):**
 - Tiempo dedicado a tareas productivas (cpu, entrada/salida)
 - $ts = tCPU + tE/S$
- **Tiempo de espera (te):**
 - Tiempo que un proceso pasa en colas de espera (cola de entrada + preparado, esperando a su turno)

Medida de prestaciones



te	Tiempo de espera, en cola de entrada, o cargando, o preparado
tCPU	Tiempo de procesamiento (activo)
tE/S	Tiempo de E/S (bloqueado)

Medida de prestaciones

- **Tiempo en máquina, o tiempo de retorno (tq)**
 - Tiempo que está un proceso en el sistema. Instante final (tf) menos instante inicial (ti)
- **Tiempo de respuesta (tiempo en máquina para procesos interactivos)**
 - intervalo de tiempo entre el instante en que el usuario da una orden y el instante en que se visualiza la respuesta.
 - Indica el retardo experimentado
- **Coeficiente de respuesta (Rn):**
 - Razón entre tiempo de retorno y tiempo de CPU
 $R_n = tq/tCPU$
 - Representa una medida de la lentitud del computador en responder al usuario.

Medida de prestaciones

- **Tiempo en máquina, o tiempo de retorno (tq)**
 - Tiempo que está un proceso en el sistema. Instante final (tf) menos instante inicial (ti)
- **Tiempo de respuesta (tiempo en máquina para procesos interactivos)**
 - intervalo de tiempo entre el instante en que el usuario da una orden y el instante en que se visualiza la respuesta.
- **Tiempo de retorno normalizado (tn):**
 - Razón entre tiempo de retorno y tiempo de servicio
$$tn = tq/ts$$
 - Indica el retardo experimentado
- **Coeficiente de respuesta (Rn):**
 - Razón entre tiempo de retorno y tiempo de CPU
$$Rn = tq/tCPU$$
 - Representa una medida de la lentitud del computador en responder al usuario.

Modos de procesamiento

- **Procesos multiprogramación:** hay distintos procesos en memoria principal y el SO va asignando el procesador alternativamente en el tiempo. A veces se les denomina sistemas multitarea.
- **Sistema multiusuario:** sistema de multiprogramación que prevé el uso concurrente de distintos usuarios.

Modos de procesamiento

- **Procesamiento por lotes (cola serie).**

- El planificador selecciona trabajos de la cola serie, cargándolos en memoria conforme haya sitio en ella, e inicia su ejecución.
- Objetivo: maximizar uso del procesador (rendimiento del procesador)

- **Sistema de tiempo compartido (o interactivo).**

- Sistema multiprogramación y multiusuario que gestiona procesos interactivos, a través de los cuales el usuario interactúa directamente con el computador casi de forma inmediata, dándole la ilusión de que está trabajando el sólo con el computador.
- Objetivo: minimizar el tiempo de respuesta.

Modos de procesamiento

- **Sistema de tiempo real:**

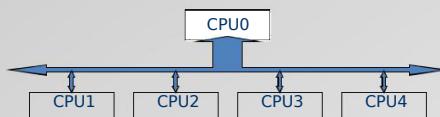
- Necesariamente el tiempo de respuesta debe ser menor de un valor predeterminado (control industrial, sistemas embebidos, redes, control de vuelo, aplicaciones militares, simuladores en tiempo real).

- **Sistema para proceso de transacciones:**

- Sistema de tiempo real que realiza conjuntos de operaciones como una unidad de trabajo atómica.
- Aplicación típica: uso de bases de datos
 - cajeros de bancos, reservas de billetes de avión, por ejemplo – AMADEUS).
- Aseguran que las transacciones se completen y se hagan rápidamente.

Sistema de multiprocesamiento

- Sistemas con 2 o más procesadores (“núcleos”) trabajando en paralelo, compartiendo la memoria principal y otros recursos.
- Cada procesador ejecuta una hebra diferente.
 - Multiprocesamiento asimétrico:
 - un procesador controla al resto (alguna versión de UNIX).



- Multiprocesamiento simétrico:
 - todos los procesadores son iguales. (Ejemplo: Windows-NT). Incremento lineal de prestaciones.



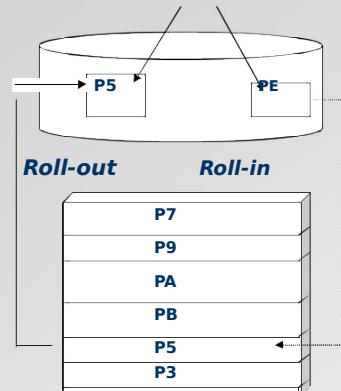
Intercambio memoria principal/disco (*swapping*)

- En un sistema en tiempo compartido (por ejemplo, que controle los cajeros de usuarios de un banco), hay que atender necesariamente a todos los que se conecten en un momento dado.
 - Puede ser que los procesos de todos los usuarios no quepan en ese momento en la memoria principal, con lo que no se pueden ir ejecutando concurrentemente en multiprogramación.
 - Solución: Intercambio memoria principal/disco.

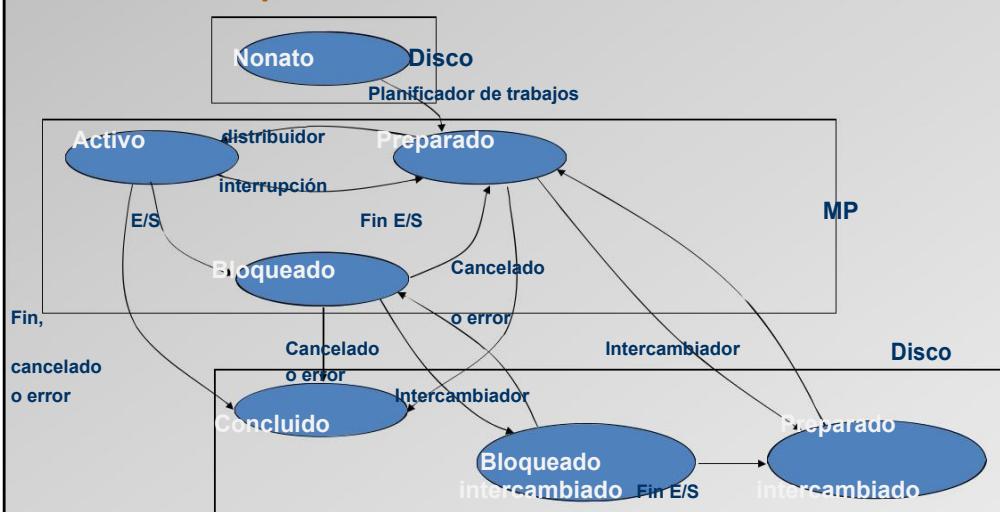
Intercambiador

- Pasar a disco (“roll-out”), muy rápidamente con el DMA, procesos bloqueados o listos de MP
- Cuando un proceso en disco se desbloquee, o se le vaya a dar turno, rápidamente se le carga en la memoria principal (“roll-in”).
- Es gestionado por el planificador a medio plazo del SO o intercambiador.
- El proceso en disco se dice que está suspendido

Procesos suspendidos



Estados de un proceso



Planificadores

- **Hasta ahora sobre “planificación” hemos visto:**
 - Módulos del SO que implementan un algoritmo de planificación
 - Tres planificadores de programas/procesos.
 - **largo plazo o planificador de trabajos:** selecciona los programas que inician su ejecución (desde la cola batch, por ejemplo)
 - **medio plazo o intercambiador:** selecciona qué procesos se transvasan (suspenden) a disco, y cuales se recuperan de disco a MP.
 - **corto plazo o distribuidor:** selecciona entre los procesos preparados el siguiente a ejecutar.

Estrategias de asignación del procesador a los procesos

• **No apropiativa:**

- El sistema operativo no puede dar el turno a otro proceso, a no ser que el que está en ejecución:
 - Termine
 - Se bloquee (operación E/S u otro servicio)
 - Ceda el procesador a otro proceso (multiprogramación cooperativa)

Ej. MS-DOS hasta Windows 3.x y Mac OS

• **Apropiativa:**

- el SO puede interrumpir en cualquier momento al proceso activo, pasándolo a preparado, para dar turno a otro según un algoritmo de planificación.

Ej. Windows 95[▲], OS/2, Unix, Linux

Puntos de decisión de planificación

- **Momentos en los que se puede decidir la planificación de un proceso:**
 1. Cuando un proceso se bloquea en espera de un evento
 - Realización de una llamada al sistema para una E/S
 2. Cuando se produce una interrupción
 - Interrupción del reloj
 - Interrupción de fin de E/S
 3. Cuando llega un proceso nuevo
 4. Fin de proceso
- **Planificación no apropiativa: 1 y 3**
 - Windows95, MacOS anteriores a versión 8
- **Planificación apropiativa: 1, 2 y 3**

Algoritmos de planificación de la CPU

- **Primero en llegar, primero en servir (FCFS)**
- **Primero el proceso más corto (SPN)**
- **Turno rotatorio (o cíclico)**
- **Derecho preferente (por prioridades)**
- **Menor tiempo restante (SRT)**
- **Mayor coeficiente de respuesta (HRRN)**
- **Realimentación**

Algoritmos de planificación de la CPU

- **Primero en llegar, primero en servir (FCFS):**
 - De acuerdo con el instante en que nacen (no apropiativo).
- **Primero el proceso más corto (SPN):**
 - Los que tienen menos tiempo de CPU, antes (no apropiativo).
 - Solamente se puede aplicar si se conoce de antemano la duración de cada trabajo
 - Posibilidad de inanición:
 - Si continuamente llegan trabajos cortos, los trabajos largos nunca llegan a ejecutarse
 - El SO Windows es multitarea con derecho preferente

Algoritmo de turno rotatorio (Cíclico, *Round-Robin*)

- Mantiene una cola FIFO con los procesos listos para ser ejecutados
- Un proceso recibe el procesador durante una fracción de tiempo (quantum)
- Un proceso regresa a la cola de preparados cuando:
 - Expira su rodaja de tiempo
 - Se produce el evento que lo llevó a la cola de bloqueados
- Un proceso pasa a la cola de bloqueados cuando:
 - Pasa a esperar un evento
- Algoritmo apropiativo
- A veces se asigna una prioridad, de forma que se baja la prioridad de cada proceso, cada vez que se le da turno.
- Se debe tener en cuenta que cada cambio de contexto genera retraso
 - quantum de tiempo >> tiempo para cambio de contexto

Algoritmos de derecho preferente

- Cada proceso tiene una prioridad asignada
- Se selecciona primero los procesos más prioritarios
- Alternativas:
 - Prioridades fijas ↗ problema de inanición
 - Solución: mecanismos de envejecimiento (bajar la prioridad de cada proceso, cada vez que se le da turno)
- Derecho preferente apropiativo: siempre se ejecuta el proceso de mayor prioridad completamente, y no se interrumpe hasta que finalice, se bloquee, o llegue otro de mayor prioridad. Muy utilizado en sistemas de tiempo real.

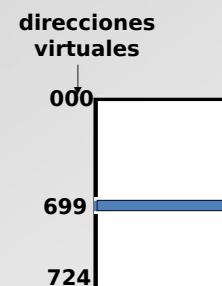
Contenidos

- Definición
 - Objetivos y funciones
 - Contexto
 - Arranque de un computador
- Gestión de recursos:
 - Gestión del procesador
 - Gestión de la memoria
 - Gestión de E/S
 - Gestión de archivos
- Sistemas operativos más conocidos
- Seguridad



Direcciones virtuales

- Un programa máquina es un conjunto ordenado de instrucciones en código máquina, que, en el momento de cargarse en la MP, encajarán en n palabras de memoria que pueden numerarse correlativamente de la 0 a la n-1. Las direcciones del programa se denominan direcciones virtuales o lógicas o relativas (dv).

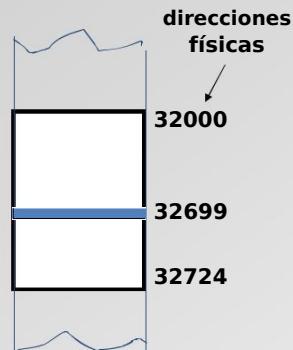


Direcciones físicas

- Los programas se cargan en la MP a partir de una posición de memoria arbitraria, en función de las zonas libres de ella. Esa dirección se denomina dirección base (DB). Las direcciones reales donde se carga el programa se denominan direcciones físicas (df), verificándose:

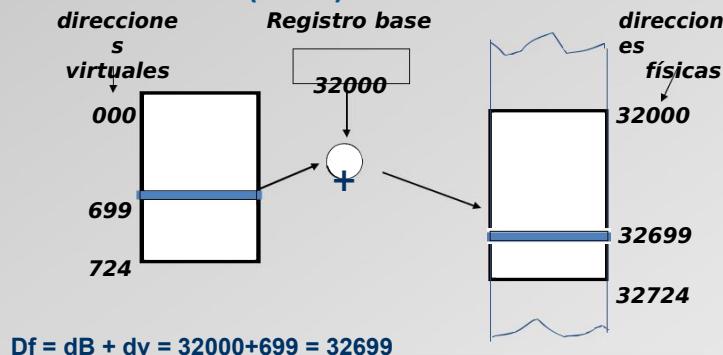
$$df = DB + dv,$$

para todo $0 \leq dv \leq n-1$



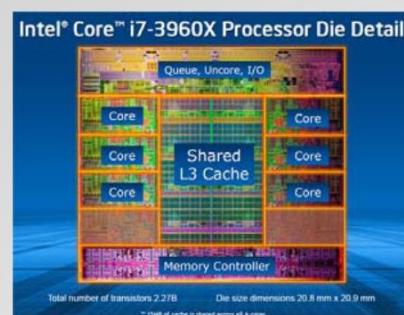
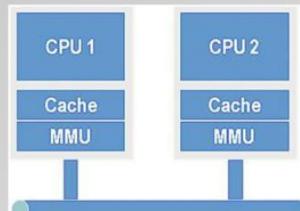
Transformación de direcciones virtuales a físicas

- Se suele realizar con ayuda de un **registro base** (que se carga previamente con la dirección base) que se encuentra o en la propia CPU o en unos ctos. específicos denominados **Unidad de Gestión de Memoria (MMU)**



La MMU se suele incluir en el chip del procesador

- En la actualidad, la MMU suele incluirse en el mismo chip del microprocesador, y realiza transformaciones entre direcciones virtuales y físicas que antiguamente se hacían por software

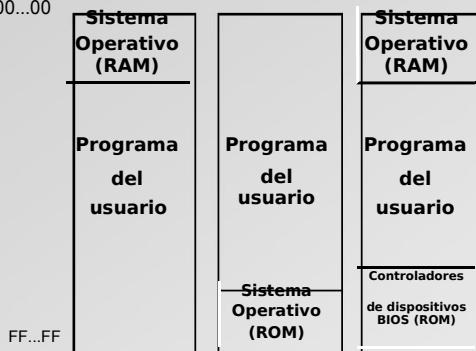


Ubicación del SO en la memoria

- **En los SO de monoprogramación**

– la MP suele organizarse de diversas formas:

- El SO puede ocupar las primeras 00...00 posiciones o las últimas (ROM)
- Parte puede estar en RAM en direcciones bajas y otra parte en ROM en direcciones altas.
- Los PCs actuales siguen el 2º modelo y en ellos la ROM se denomina: **Sistema básico de Entradas/Salidas (BIOS)**



Ubicación de procesos y datos en la memoria interna

- **En multiprogramación**

– el SO asigna espacio de memoria a los programas o procesos, de acuerdo con las zonas libres de memoria existentes en cada momento.

– Dependiendo de las posibilidades del SO, la asignación puede realizarse de las siguientes formas:

- » Particiones estáticas
- » Particiones dinámicas
- » Segmentación
- » Paginación
- » Memoria virtual

Particiones estáticas

- La memoria se divide en cierto número de particiones o zonas, cada una de las cuales contendrá un proceso.
- Las direcciones de base son las direcciones de comienzo de cada partición.
- El tamaño de la partición es determinado por el SO.
- EJEMPLO: memoria de 2MB, con particiones de 256KB.

Partición		
0	SO	000000
1	SO	040000
2	Proceso 3	080000
3	Proceso 27	0C0000
4	Proceso 43	100000
5	Proceso 5	200000
6	Proceso 7	300000
7	Proceso 0	400000
		4FFFFF

→ 7
Fragmento libre



Gestión de particiones estáticas

- El SO mantiene una tabla en la que cada fila corresponde a una partición, conteniendo la posición base de la partición, su tamaño (no todas las particiones tienen por qué ser iguales) y el estado de la partición (ocupada o no ocupada).

Partición x	Posición base	Tamaño	Estado
-------------	---------------	--------	--------

- El planificador de trabajos, una vez que una partición está libre, hace que se introduzca el programa de máxima prioridad que haya en la cola de espera y que quepa en dicha partición.



Particiones dinámicas

- Los programas son introducidos por el SO inicialmente en posiciones consecutivas de memoria, no existiendo por tanto particiones predefinidas.

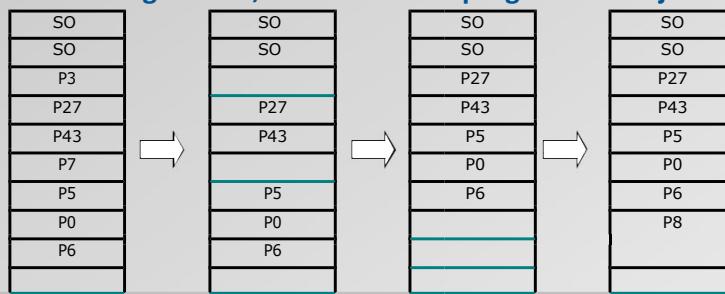
SO
SO
P3
P27
P43
P7
P5
P0
P6

Gestión por el SO de las particiones dinámicas

- **Tabla de procesos**, en la que cada línea contiene el identificativo de trabajo, el espacio que ocupa, y la dirección base.
- **Tabla de fragmentos libres**.
 - El planificador de trabajos, consultando esta tabla, introduce en la MP los programas que quepan en los fragmentos libres; y actualiza las dos tablas

Reubicación de procesos

- Al ir acabando de ejecutarse los programas, el nº de fragmentos libres crecerá y el espacio de cada uno de ellos disminuirá, llegando un momento en que el porcentaje de memoria aprovechado es muy reducido.
- El problema se resuelve haciendo una **compactación** que agrupa todos los fragmentos, **reubicando los programas en ejecución**.



Fundamentos de Sistemas Operativos

73 / 162



Segmentación

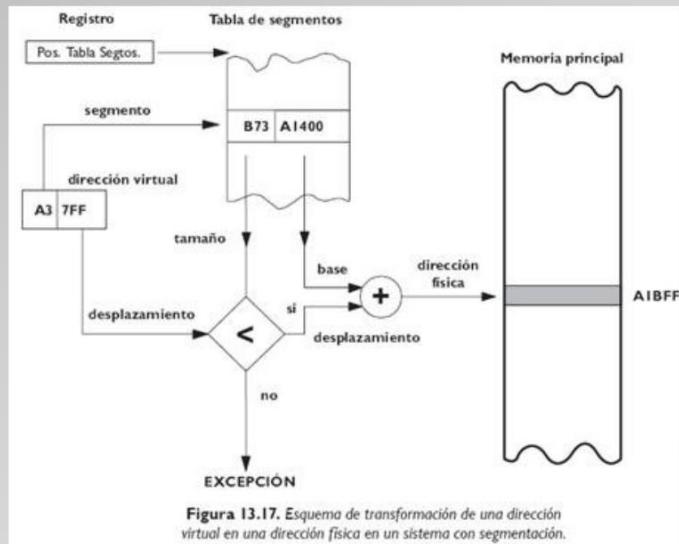
- El programa se considera dividido en segmentos:
 - Segmento de código (programa en lenguaje máquina)
 - Segmento de pila
 - Segmento(s) de datos
- La gestión la realiza el SO, como con las particiones dinámicas, sólo que cada partición, en vez de corresponder a un programa completo, corresponde a un segmento.
- El S.O. mantiene, por cada proceso, una tabla donde se especifica la dirección base de cada segmento, y su capacidad.
- La segmentación permite que ciertos procesos puedan compartir código (rutas, etc.) o datos comunes sin necesidad de estar duplicados en MP.

dcc. virtual		Tabla segmen.	dcc. física	
seg	offset	dB, C _{máx}	df	Si offset ≤ C _{máx} : df = dB + dv

Fundamentos de Sistemas Operativos

74 / 162





Paginación

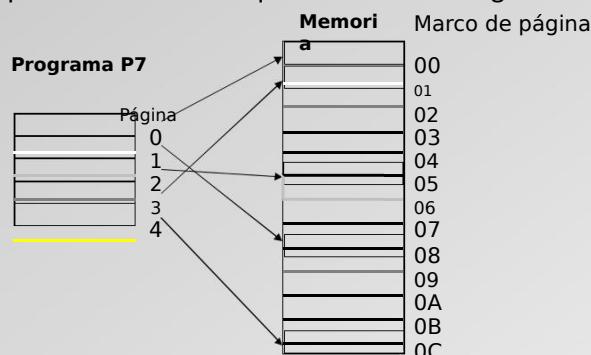
- Los programas se consideran divididos en zonas consecutivas, (**páginas**).
 - Cada página es de tamaño fijo (512 B a 64 KB, por lo general).
 - Se identifican dentro del programa, con un número correlativo.
- La MP se estructura en **marcos de página** de igual capacidad que las páginas del programa.
 - Cada marco se identifica con un número correlativo.
 - Si el tamaño de la MP fuese de 1MB, y el tamaño del marco 4KB, habría 256 marcos de página (del 00 al FF).

Programa	Página
a	0
	1
	2
	3
	4

Memoria	Marco de página
a	00
	01
	02
	03
	04
	05
	06
	07
	08
	09
	0A
	0B
	0C

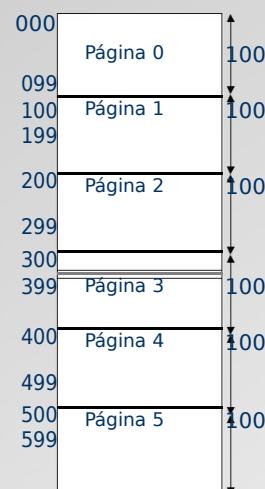
Gestión de la paginación

- El fundamento de la paginación está en que no es necesario que un programa se almacene en posiciones consecutivas de memoria.
- Las páginas se almacenan en marcos de página libres independientemente de que estén o no contiguos.



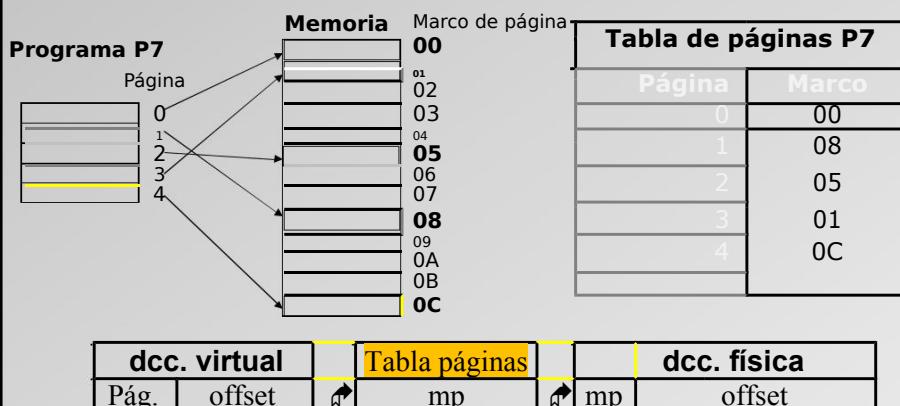
Gestión de las direcciones de acceso

- Una dirección virtual (una dirección del programa) puede considerarse formada por el **número de página** (bits más significativos, MSB) y un **desplazamiento dentro de la página** (bits menos significativos, LSB).
- Por ejemplo, supongamos que estamos en base 10, y las páginas son de 100 posiciones. Entonces la dirección 328, corresponderá a la página 3, y, dentro de ella tendrá una posición relativa (desplazamiento) de 28 posiciones.



Tablas de páginas

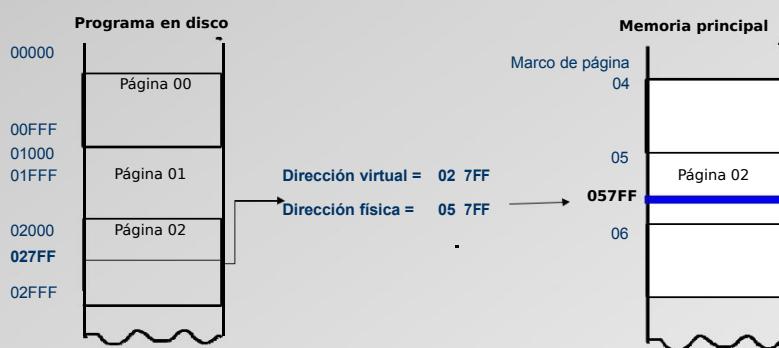
- Cada proceso tiene asociado una **Tabla de Páginas**, que, sencillamente, indica el marco de página donde se encuentra almacenada la página correspondiente.



Transformación a dirección física

- La **dirección física** se obtiene a partir de la dirección virtual, consultando la **tabla de páginas** del proceso, y concatenando el **marco de página** con el **desplazamiento**.

Tabla de páginas	
Pág.	Marco
00	00
01	08
02	05
3	01
4	0C



Gestión de la paginación por el SO

- **El SO mantiene tres tipos de tablas:**

- **Tabla de páginas.** Hay una por proceso, y contiene el marco de página donde se encuentra cada una de sus páginas. La longitud de cada tabla es variable, dependiendo de la capacidad (nº de páginas) del proceso.
- **Tabla de marcos de memoria.** Contiene tantas filas como marcos. Se indica el identificativo del proceso que está en cada marco, y, en su caso, si está libre.
- **Tabla de procesos.** Cada fila contiene información referente a cada proceso en ejecución. Se indica su tamaño, dirección de memoria, dónde se encuentra su tabla de páginas, y el estado y situación del proceso

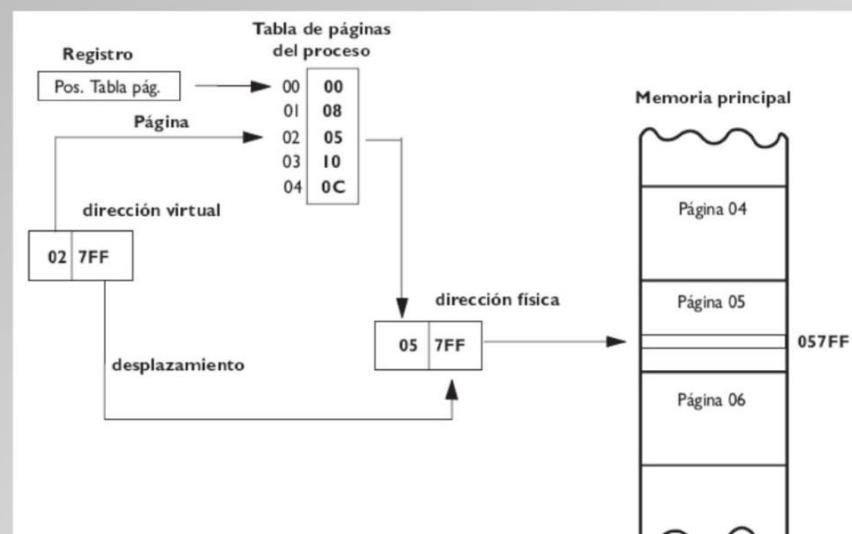


Figura 13.20. Esquema de transformación de direcciones en un sistema con paginación.

Memoria Virtual

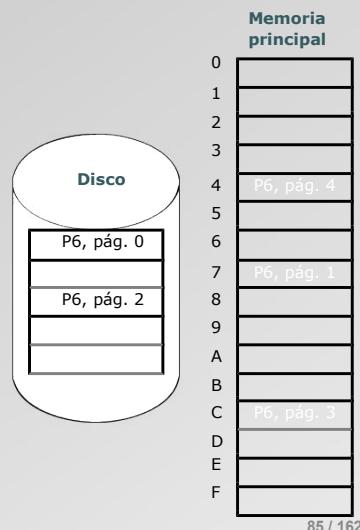
- **La memoria virtual (MV) permite**
 - ejecutar programas de capacidad superior a la memoria que físicamente tiene el computador.
 - aumentar el número de procesos en la MP, en ejecución concurrente.
- **En definitiva presenta al usuario una MP aparente mayor que la memoria física real.**
- **Para implementar la memoria virtual suele utilizarse una de las siguientes técnicas:**
 - gestión de memoria por páginas,
 - gestión de memoria segmentada,
 - gestión de memoria segmentada/paginada

Principios en los que se fundamenta la MV

- **Las instrucciones de un programa que se ejecutan sucesivamente (en un corto intervalo de tiempo) están en direcciones muy próximas (**principio de localidad temporal**),**
- **Los programas suelen estar redactados con gran linealidad; es decir, no suelen abundar los saltos entre posiciones de memoria distantes (**principio de localidad espacial**).**

Concepto básico de la MV

- En un sistema de MV se tiene en disco el proceso completo, que se considera por la CPU troceado en páginas o segmentos, dependiendo del método.
- Únicamente se cargan en la MP la página o páginas (o segmentos, en su caso) que en ese momento intervengan en el proceso, intercambiando páginas entre disco y MP cuando sea necesario.



Gestión de MV con paginación

- En el caso de que el control de un programa llegue a una instrucción (o requiera un dato) de una página que no esté en memoria, se dice que se produce un fallo de página, y el SO debe cargar en memoria la página correspondiente.

Tablas de páginas para gestión de la MV

- El SO hace la gestión de la memoria virtual con dos tablas:

- **Tabla de páginas**: el SO crea y mantiene una por cada proceso. Cada fila corresponde a una de sus páginas y contiene:
 - Posición en el disco donde comienza la página.
 - Marco de página en el que está la página, y
 - Un bit que indica la situación de la página: 0 si está en MP, 1 si no está.

página	Posición disco	Marco de página	Ubicación en memoria
0	7ABC	-	1
1	CA73	7	0
2	4BC9	-	1
3	573C	C	0
4	A340	4	0

Tabla de marcos de página para gestión de la MV

Tabla de marcos de página

Marco pág.	Proceso/Página	Estado	Cambio
0	6,03	0	1
1	3,05	0	0
2	4,BC	0	0
3	6,04	0	0
4	0,A3	0	1
5	0,A4	0	0
6	Libre	1	0
7	6,0	0	0

Tabla de marcos de página para gestión de la MV

- Cada fila corresponde a un marco de página, y contiene lo siguiente:
 - Proceso/página que está en el marco
 - Estado del marco, 0: ocupado; 1:libre
 - Cambio: bit para indicar si, desde que se ha cargado la página desde disco por última vez, se ha modificado (1) o no (0)
 - Otra información adicional, dependiente del algoritmo de reemplazo de página.

Tabla de marcos de página			
Marco pág.	Proceso/Página	Estado	Cambio
0	6,03	0	1
1	3,05	0	0

Ejemplo 1 de gestión de la MV

- El proceso P6 hace referencia a una instrucción o dato de la página 0. El SO consulta la tabla de páginas P6 y detecta que no está en MP (ubicación=1) (fallo de página).
- El SO busca en la tabla de marcos de página un marco libre (estado=1), y encuentra que el marco 6 está libre.
- Entonces se carga la página solicitada, que según la tabla de páginas se encuentra en disco a partir de la posición 7ABC.
- El SO actualiza las tablas de acuerdo con la nueva situación.

Tabla de páginas P6			
(pág)	Posición disco	Marco de página	Ubicación (0 en memoria)
0	7ABC7	6-	01
1	CA73	7	0
2	4BC9	-	1
3	573C	C	0
4	A340	3	0

Tabla marcos de página			
Marco de página	Proceso/Pag	Estado	Cambio
0	6,03	0	1
1	3,05	0	0
2	4,BC	0	0
3	6,04	0	0
4	0,A3	0	1
5	0,A4	0	0
6	libre	1	0
7	6,1	0	0

Ejemplo 2

- Se solicita Página 2

- no está en MP (ubicación=1),
- no hay ningún marco libre (est=1) => Hay que eliminar una pág. de MP
- Suponga que el algoritmo de reemplazo decide eliminar **pág.6.4**
- Como bit cambio =0 => no se ha modificado desde que se introdujo en MP
- Se carga **Página 2** en **MP 3**

Tabla de páginas P6			
(pág)	Posición disco	Marco de página	Ubicación
0	7ABC	6	0
1	CA73	7	0
2	4BC9	-	1
3	573C	C	0
4	A340	3	0

Tabla de marcos de páginas			
marco	Proceso/pág.	Estado	Cambio
pág 0	6,03	0	1
1	3,05	0	0
2	4,BC	0	0
3	6,04	0	0
4	0,A3	0	1
5	0,A4	0	0
6	6,00	0	0
7	6,01	0	0

Ejemplo 2

- Se solicita Página 2

- no está en MP (ubicación=1),
- no hay ningún marco libre (est=1) => Hay que eliminar una pág. de MP
- Suponga que el algoritmo de reemplazo decide eliminar **pág.6.4**
- Como bit cambio =0 => no se ha modificado desde que se introdujo en MP
- Se carga **Página 2** en **MP 3**

Tabla de páginas P6			
(pág)	Posición disco	Marco de página	Ubicación
0	7ABC	6	0
1	CA73	7	0
2	4BC9	3	0
3	573C	C	0
4	A340	3	0

Tabla de marcos de páginas			
marco	Proceso/pág.	Estado	Cambio
pág 0	6,03	0	1
1	3,05	0	0
2	4,BC	0	0
3	6,04	0	0
4	0,A3	0	1
5	0,A4	0	0
6	6,00	0	0
7	6,01	0	0

Ejemplo 2

- Se solicita Página 2

- no está en MP (ubicación=1),
- no hay ningún marco libre (est=1) => Hay que eliminar una pág. de MP
- Suponga que el algoritmo de reemplazo decide eliminar **pág.6.4**
- Como bit cambio =0 => no se ha modificado desde que se introdujo en MP
- Se carga **Página 2** en **MP 3**

Tabla de páginas P6			
(pág)	Posición disco	Marco de página	Ubicación
0	7ABC	6	0
1	CA73	7	0
2	4BC9	3	0
3	573C	C	0
4	A340	-	1

Tabla de marcos de páginas

marco	Proceso/pág.	Estado	Cambio
pág			
0	6,03	0	1
1	3,05	0	0
2	4,BC	0	0
3	6,04	0	0
4	0,A3	0	1
5	0,A4	0	0
6	6,00	0	0
7	6,01	0	0

Ejemplo 2

- IMPORTANTE:

- Si el bit de cambio del mp fuese 1, antes de memorizar la página 6,02, habría que actualizar la copia de disco de la página 6,04 (que está en el clúster A304)

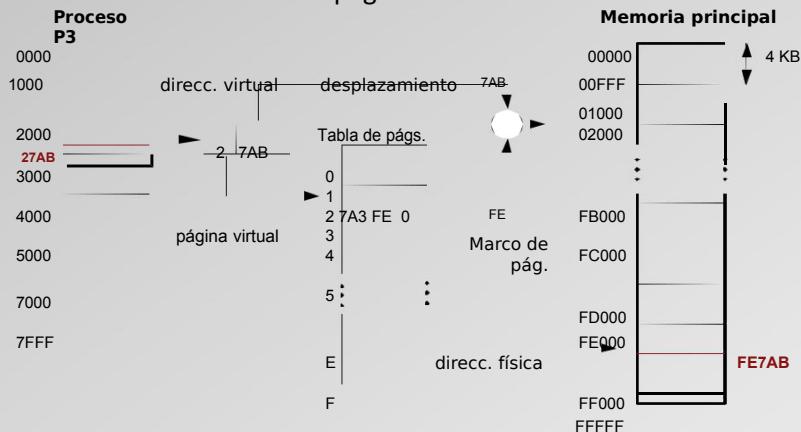
Tabla de páginas P6			
(pág)	Posición disco	Marco de página	Ubicación
0	7ABC	6	0
1	CA73	7	0
2	4BC9	3	0
3	573C	C	0
4	A340	3	0

Tabla de marcos de páginas

marco	Proceso/pág.	Estado	Cambio
pág			
0	6,03	0	1
1	3,05	0	0
2	4,BC	0	0
3	6,04	0	1
4	0,A3	0	1
5	0,A4	0	0
6	6,00	0	0
7	6,01	0	0

Gestión de la MP

- Obtención de la dirección física (en la MMU) a partir de la virtual, consultando la tabla de páginas:



Gestión de reemplazo de páginas

- **¿Qué páginas deben estar en memoria?**

- Criterio de carga en memoria: decide qué páginas se cargan en memoria:

- Paginación por demanda: una página sólo se lleva a MP cuando se hace referencia a una dirección de ella.
- Paginación previa: Se cargan en MP más páginas de las referenciadas, usualmente consecutivas (para aprovechar la localidad espacial, y para ganar en velocidad).

- Criterio de reemplazo de página: ¿qué páginas se sacan de la MP para introducir otra? Los algoritmos de sustitución de página más conocidos son: FIFO, LRU, NRU, Algoritmo del reloj

Algoritmos de reemplazo ...

- **FIFO:**

- se sustituye la página que lleve más tiempo en memoria.
- Implementación: se incluye en un campo de la tabla de marcos de página, el instante de tiempo en que se carga.

- **LRU:**

- Se sustituye la página que lleve más tiempo sin usar.
- Se basa en el principio de localidad temporal ya que supone que la página utilizada hace mayor tiempo, es menos probable que se use próximamente.
- Implementación: con un contador, C, asociado a cada marco de página, que se incrementa cada vez que se efectúe un acceso a memoria; y se pone a 0 cada vez que se accede a su página.

... Algoritmos de reemplazo

- **NRU:**

- Se sustituye la página que menos se haya utilizado recientemente.
- Implementación: se utiliza una variable R (“referido”) asociada a cada marco, que se pone a 1 cuando se referencia el marco. De vez en cuando (20 ms, por ejemplo) se ponen a cero todas las variables R.

- **Algoritmo del reloj:**

- Un puntero va recorriendo los distintos marcos de página. En cada fila de la tabla de marcos se incluye un bit (0 ó 1) de referido (R). Cuando se introduce una página, el bit se pone a 0 (R=0). Si se accede a un marco, se hace R=1, y no se avanza el puntero. Si se produce un fallo de página: se analiza el R al que apunta el puntero:
 - Si R=0, se reemplaza la página y se avanza una posición el puntero (permaneciendo R=0).
 - Si R=1, se cambia a R=0, y así sucesivamente se avanza el puntero hasta encontrar un R=0.

Ejemplos de algoritmos de reemplazo

- Memoria con 3 marcos de página, e inicialmente vacía.
- Se hacen referencia sucesiva a las siguientes páginas:

2, 3, 2, 1, 5, 2, 4, 5, 3, 2, 5, 2

Ejemplos de algoritmos de reemplazo

FIFO	
LRU	
RELOJ	

Contenidos

- **Definición**
 - Objetivos y funciones
 - Contexto
 - Arranque de un computador
- **Gestión de recursos:**
 - Gestión del procesador,
 - Gestión de la memoria
 - Gestión de E/S 
 - Gestión de archivos
- **Sistemas operativos más conocidos**
- **Seguridad**

Gestión de entradas/salidas

- **El SO administra los periféricos (unidades de disco, cinta magnética, etc., y dispositivos de entrada/salida), con los siguientes objetivos básicos:**
 - Posibilitar que los recursos citados puedan ser compartidos eficientemente por distintos procesos. Para ello suele utilizarse la técnica de dispositivos de entrada/salida virtuales, y
 - Hacer lo más transparente posible al usuario las características particulares del hardware que utiliza. Para ello el SO suele disponer de módulos de gestión de entradas/salidas, y de gestión y administración de archivos

Gestión de entradas/salidas

- Si se utiliza un periférico desde un programa-máquina, las instrucciones correspondientes tienen que codificarse de acuerdo con las características concretas del periférico.
 - El SO permite trabajar con los dispositivos de una forma abstracta (independientemente de las características de los dispositivos). Ejemplo en C:

```
numero = read (nombreachivo , zona , numB) ;
```

- El SO incluye gestores o manejadores de periféricos (*drivers*), que son rutinas de E/S encargadas de controlar los dispositivos. Estos gestores son los únicos que tienen en cuenta las peculiaridades concretas de los dispositivos.
 - Cuando instalamos una impresora, p.e., tenemos que incluir el programa controlador correspondiente
- La rutina de entrada/salida realiza las comprobaciones necesarias sobre la operación que se va a realizar, y hace la petición de servicio al gestor del periférico correspondiente, que será el que efectúe la operación.

Gestión de entradas/salidas

- Las operaciones de E/S se entienden mejor utilizando un modelo conceptual por capas



Gestión de entradas/salidas

- **Nivel de procesos de aplicaciones.**

- Corresponde a los procesos de aplicaciones que tienen operaciones de E/S. Se hace una llamada al SO que lanza a ejecución el gestor del periférico. El SO puede gestionar un dispositivo de E/S de una de las tres formas siguientes:
 - **Dispositivos asignados.** Consiste en asignar ciertos dispositivos (un teclado, una pantalla o una impresora; por ejemplo) a un usuario o proceso durante la duración del trabajo.
 - **Dispositivos compartidos.** Es como se asignan los dispositivos que pueden compartirse concurrentemente por varios procesos (discos), y
 - **Dispositivos virtuales ("spooler").** Se consigue que dispositivos en principio asignables (impresora, por ejemplo) puedan compartirse, optimizándose el rendimiento del sistema. Es el caso de una impresora de alta velocidad, a compartir por distintos usuarios.

Gestión de entradas/salidas

- **Nivel de software de E/S independiente del dispositivo**

- En este nivel se incluyen funciones de E/S comunes a todos los dispositivos (abrir, leer, escribir, cerrar) En UNIX y Windows NT los dispositivos se utilizar como archivos especiales. Tipos de cometidos:
 - A partir de la denominación simbólica del archivo se localizan las tablas que contienen los atributos del dispositivo (dirección del controlador, dirección del sector, etc.)
 - Protección de datos (contraseña de accesos, y derechos de acceso: *leer/leer-escribir*, etc.).
 - Asignación y liberación de dispositivos dedicados
 - Informes de errores, etc.

Gestión de entradas/salidas

- **Nivel de organización física**

- Este nivel tiene en cuenta las características específicas de cada tipo de periférico.
- Funciones de alto nivel del periférico (organización RAID, etc.).
- Almacenamiento intermedio. Ejemplo:
 - Cache de disco: Una vez localizado el cilindro se puede leer el cilindro completo a la velocidad de rotación del disco, y su contenido almacenarlo en la **caché de disco**, de forma que si el programa del usuario ordena la lectura sucesiva de sectores de un mismo cilindro no se tendrán que consumir tiempos adicionales de latencia para acceder a los distintos sectores.
- Convierte referencias lógicas a archivos y registros en direcciones físicas

Gestión de entradas/salidas

- **Disco RAM**

- Otra técnica, también basada en la idea de reducir al máximo los accesos *físicos* al disco, se denomina **disco RAM**.
 - consiste en llevar a memoria un conjunto de bloques ("clusters") del disco, mayor cuanto mayor sea la capacidad de la MP, y los procesos en lugar de intercambiar la información con el disco físico lo hacen con su imagen en MP.
 - Cuando acaban las operaciones con el disco (se cierra el archivo), se actualiza el disco físico. Este procedimiento se puede gestionar en forma similar a la memoria virtual. Podemos hablar aquí también de **periférico virtual**, en el sentido de que en la MP tenemos un **disco virtual** (en realidad sólo una parte de él).

Gestión de entradas/salidas

- **Nivel de gestión software de dispositivo hardware**
 - Planificación y control
 - Generación de las instrucciones (patrones de bits) concretas para interactuar con el hardware
 - Atención de interrupciones

Gestión de entradas/salidas

- Ejemplo de generación de las instrucciones concretas para el controlador físico.
 - En un servidor de archivos (disco), al presentarse al gestor correspondiente una operación de leer un sector, el gestor debe comprobar si el disco está arrancado (y sino arrancarlo), posicionar el brazo en el cilindro correspondiente, inicializar el DMA, transmitir el marco de página de información, etc.
 - En realidad los gestores de periféricos (*drivers*) son los únicos programas que deben tener en cuenta las peculiaridades concretas de los dispositivos y generar las instrucciones que éstos entienden.

Gestión de entradas/salidas

- **Atención de interrupciones**

- Cuando un proceso tiene una E/S, se inicia la operación, pasa al estado de bloqueado, y espera hasta que acabe la operación de E/S.
- En la mayoría de los dispositivos de E/S las operaciones de lectura y escritura se inician y finalizan por medio de interrupciones.
 - Cuando se produce la interrupción final, un proceso debe cambiar en la tabla de procesos el estado del proceso que ha finalizado la operación de E/S de bloqueado a preparado.

Gestión de entradas/salidas

- **Nivel de dispositivos hardware**

- Incluye los elementos físicos (controladores DMA, procesadores de E/S, tarjetas, chips, etc.) que realmente realizan las operaciones de E/S. Interactúan con el resto del computador a través de:
 - Puertos de control
 - Puertos de estado
 - Puertos de entrada
 - Puertos de salida
- Los gestores software dan las órdenes escribiendo en puertos de control; y detectan la situación del dispositivo leyendo en el puerto de estado.
- Los puertos pueden estar diseñados de forma combinada (puerto de estado/control)

Ejemplo de gestión de E/S

- **Controlador de un periférico de alta velocidad (disco).**

- Si por cada carácter que se envíase a/desde el disco
 - El controlador del periférico interrumpe a la CPU
 - Se termina instrucción en curso y se salva el contexto
 - El SO trata la interrupción y guarda/lee el carácter en la zona de E/S del periférico (en el SO)
 - Se restaura el contexto
- Si, además, se envía un carácter cada 10 nseg (100MBps) y el Sw del SO que trata la interrupción tarda 2 µseg, **no hay tiempo para tratar la interrupción por parte del SO**



PROBLEMA GRAVE

- **SOLUCIONES:**

- Buffer en periférico (transmisión de ráfagas de datos)
- Controladores de E/S
- Acceso Directo a Memoria (DMA, *Direct Memory Access*)

>>Introducción a la Informática, pp.374-376

Gestión de la E/S

Acceso Directo a Memoria (DMA)

- **Controlador DMA:**

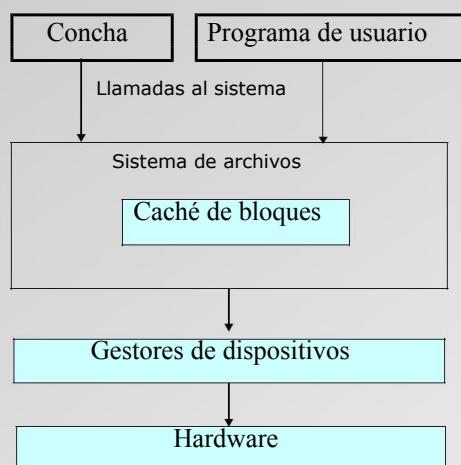
- El controlador transfiere un bloque completo de datos directamente entre su propio buffer y la MP, sin intervención de la CPU
- Sólo se genera una interrupción por bloque

- **Funcionamiento.**

- Ante una petición de E/S contra el periférico:
 - El driver software del SO configura el controlador DMA para que realice la transferencia (carga en los puertos del CDMA: direcciones origen y destino y cantidad de bytes a transferir)
 - Le ordena que realice la transferencia
 - Mientras el controlador DMA realiza la transferencia, la CPU puede ejecutar otros procesos
 - Como en la MP normalmente sólo se puede hacer una operación de lectura o escritura a la vez, el controlador DMA roba ciclos a la CPU para acceder a la MP, en los cuales la CPU no puede leer o escribir en la MP
 - El controlador de DMA interrumpe a la CPU cuando haya terminado de transferir todos los datos

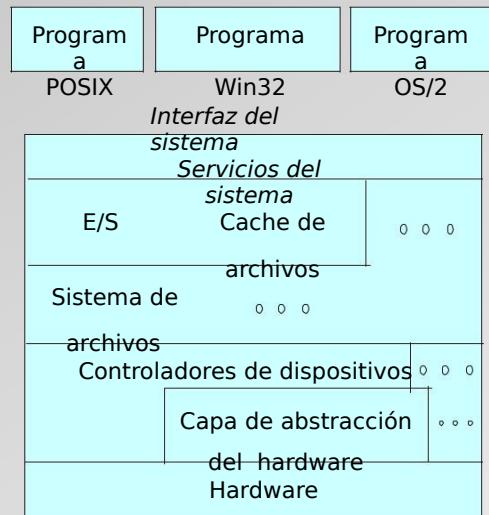
Gestión de entradas/salidas

- **Modelo de E/S del UNIX**



Gestión de E/S

- **Modelo de E/S de Windows NT**



Contenidos

- **Definición**
 - Objetivos y funciones
 - Contexto
 - Arranque de un computador
- **Gestión de recursos:**
 - Gestión del procesador,
 - Gestión de la memoria
 - Gestión de E/S
 - Gestión de archivos
- **Sistemas operativos más conocidos**
- **Seguridad**

Gestión de archivos

– Para almacenar información, físicamente se dispone de dispositivos a los que se accede por direcciones físicas. Para acceder a un sector de disco, p.e., en las primeras unidades era necesario indicar:

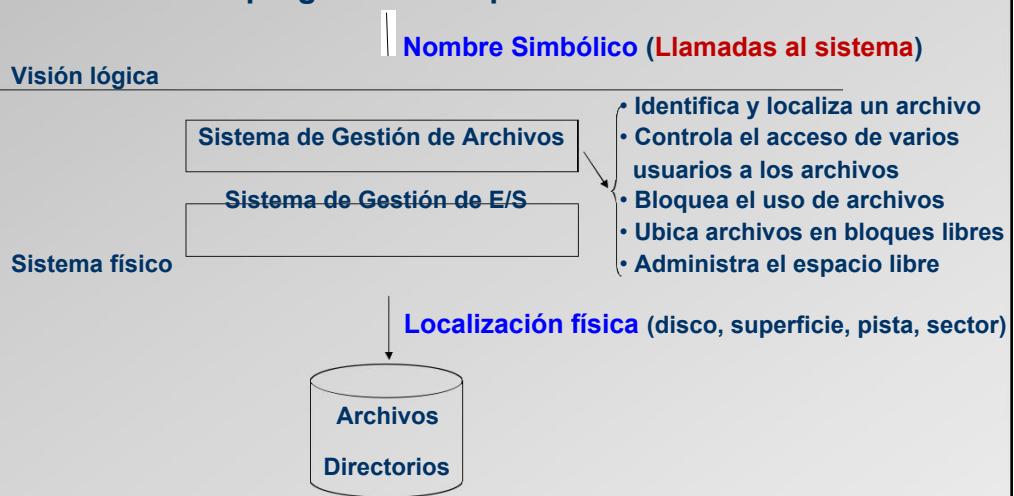
(número de unidad)/(superficie)/(pista)/(sector)

– El SO (con los conceptos de archivo y directorio) posibilita que el usuario no tenga que conocer los detalles físicos del periférico:

- Se aísla al usuario de los problemas físicos de almacenamiento.
- Dos visiones:
 - **Lógica** (en consonancia con la aplicación)
 - **Física** (en consonancia con el soporte de información)

Gestión de los archivos

Usuario: programador / aplicaciones



Gestión de archivos

- El concepto de **archivo** posibilita aislar al usuario de los problemas físicos de almacenamiento.
 - Cuando el usuario deseé referirse a un conjunto de información del mismo tipo como una unidad de almacenamiento, no tiene nada más que crear un archivo dándole el nombre que considere oportuno. Los archivos se conciben como estructuras con las siguientes peculiaridades:
 - Deben ser capaces de contener grandes cantidades de información
 - Su información debe permanecer y sobrevivir a los procesos que la generan o utilizan, y
 - Distintos procesos deben poder acceder a la información del archivo concurrentemente.

Gestión de archivos

- Cada archivo contiene su nombre (dado arbitrariamente por el usuario siguiendo unas normas preestablecidas), atributos, y los datos.
- Los atributos pueden incluir cuestiones tales como fecha y hora de creación, fecha y hora de la última actualización, bits de protección (sólo lectura, o lectura y escritura), contraseña de acceso, número de bytes por registro, capacidad máxima del archivo, capacidad actualmente ocupada, e información sobre su ubicación.
- Los datos se almacenan en el dispositivo de memoria masiva en forma de bloques.

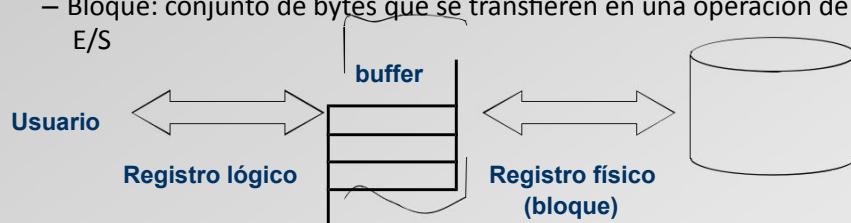
Gestión de los archivos

- **Visión lógica:**

- Archivo: sucesión de registros lógicos
 - Registro lógico: estructurado en campos
 - Campos, conjuntos de caracteres.
 - El SO puede o no facilitar diferentes tipos de reg. lógicos

- **Sistema físico:**

- Archivo: sucesión de registros físicos (disco: bloque).
- Bloque: conjunto de bytes que se transfieren en una operación de E/S



Gestión de los archivos

- **Sistema físico:**

- Tamaño de los registros físicos (bloques) Archivo:
 - bloque pequeño => poca eficiencia (más interrupciones)
 - bloque grande => fragmentación interna
- Su tamaño viene definido por el SO y por el dispositivo físico

- **Factor de blocaje:**

- nº registros lógicos por registro físico

Gestión de los archivos

- **Tabla descriptor de archivos:**
 - Almacenada en memoria y gestionada por el SO
 - Una entrada por archivo, con información del archivo:
 - Fecha y hora de creación, tamaño, bloques que ocupa,...
 - La entrada se crea al abrir un archivo y se borra al cerrarlo

Gestión de los archivos

Llamada al sistema	UNIX	Windows
Crear	open	CreateFile
Borrar	unlink	DeleteFile
Cerrar	close	CloseHandle
Leer	read	ReadFile
Escribir puntero	write	WriteFile
Mover puntero	lseek	SetFilePointer
Información	stat	GetFileAttributes

write (descrip. archivo, zona, numB)

Bloque, puntero



Gestión de los archivos

- **Puntero:**

- dirección del primer registro lógico del fichero a partir del cual se va a leer o escribir en el fichero
- En cada operación de lectura o escritura el puntero se incrementa en el número de registros lógicos leídos o escritos

- **Tipos de acceso a un fichero:**

- Acceso secuencial: los registros lógicos del fichero se procesan en orden
- Acceso directo: se accede directamente a un registro lógico cualquiera, actualizando el puntero a la dirección de ese registro

Gestión de los archivos

- **Archivo tiene asociado:**

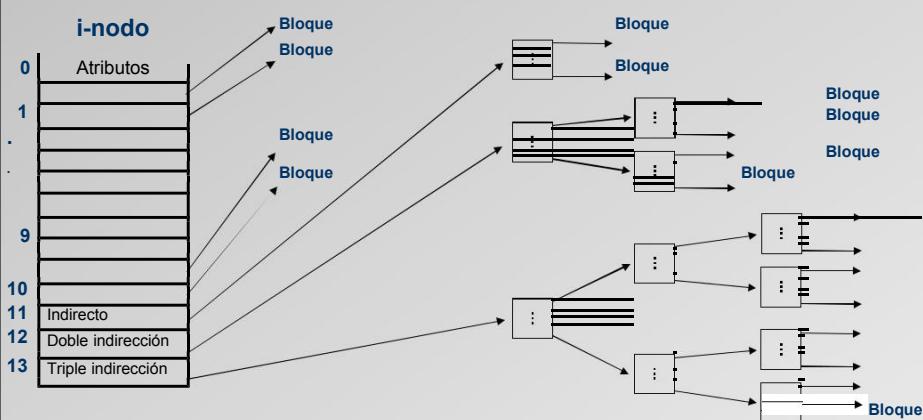
- Nombre simbólico
- Propietario
- Grupo
- Tipo (ordinario, especial, directorio)
- Atributos:
 - Fecha y hora creación
 - Fecha y hora última actualización
 - Bits de protección (propietario, grupo, todos) (RWX-RWX-RWX) (750)
- Contraseña de acceso,...
- Tamaño
- Nº de bloques → **método de asignación de espacio en disco**

Gestión de los archivos

- **Asignación de espacio en disco:**
 - **Contigua:** bloques consecutivos
 - Fragmentación externa ↗ compactación
 - **Encadenada:**
 - en cada bloque se indica la posición del siguiente
 - **Acceso secuencial**
 - **Enlazada:**
 - una tabla con tantas entradas como bloques,
 - en cada entrada el nº de bloque siguiente (FAT en MS-DOS y Windows)
 - Tabla muy grande (crece con el tamaño del disco)
 - **Nudos de índices (indexada):**
 - cada archivo una estructura (i-node en UNIX)
 - Atributos del fichero
 - Nº de los bloques asignados

Gestión de los archivos

- **Asignación de espacio en disco: Indexada (UNIX)**



Gestión de los archivos

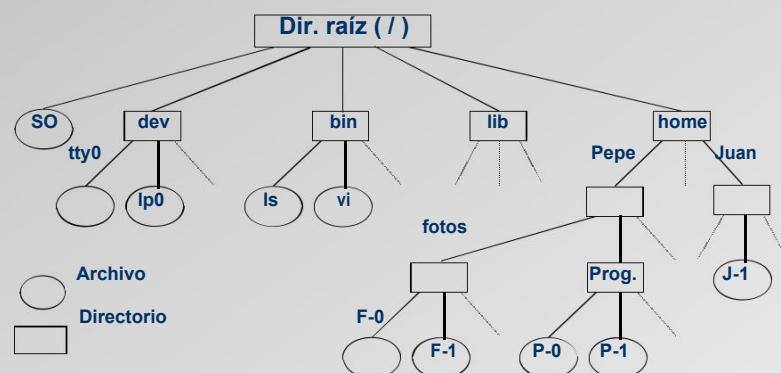
- **Directarios o carpetas:**

- Permite al usuario organizar sus archivos
- Grupo de archivos y/o subdirectorios
- Es a su vez un archivo (tabla) con una entrada por cada archivo/subdirectorio que cuelga de él, con:
 - Nombre simbólico, propietario, grupo,, nº bloques que ocupa
 - o
 - Nombre simbólico y dirección i-nodo

Gestión de los archivos

Directarios o carpetas

- **Estructura arbórea (subdirectorios)**



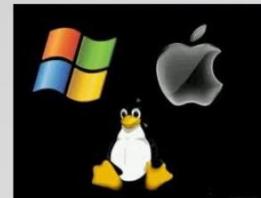
Gestión de los archivos

Directorios o carpetas

- **Ruta (path) de un archivo: lista de todos los directorios atravesados hasta llegar al archivo/directorio:**
 - Path absoluto: desde el directorio raíz (/)
 - Path relativo: desde el directorio actual
- **Si el directorio actual es Pepe:**
 - Para dirigir fichero F-1
 - /home/Pepe/fotos/F-1
 - fotos/F-1
 - Para dirigir fichero J-1
 - /home/Juan/J-1
 - ../Juan/J-1

Contenidos

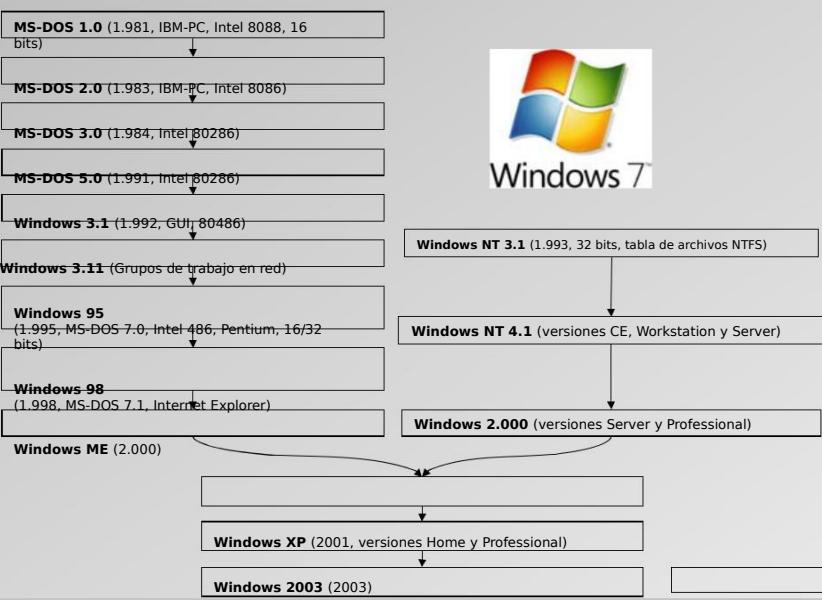
- **Definición**
 - Objetivos y funciones
 - Contexto
 - Arranque de un computador
- **Gestión de recursos:**
 - Gestión del procesador,
 - Gestión de la memoria
 - Gestión de E/S
 - Gestión de archivos
- **Sistemas operativos más conocidos** 
- **Seguridad**



Sistemas operativos de mayor difusión

- **SO de Microsoft: (85%)**

- El primero se llamó MS-DOS, y fue adoptado por IBM como SO estándar para el IBM-PC. Inicialmente era para microprocesadores de 16 bits, mono-usuario y con interfaz de línea de órdenes.
- En 1990 se comercializó Windows 3.0, con interfaz de menús.
- En 1995 salió Windows'95 con una GUI, de 16/32 bits y con memoria virtual.
- Paralelamente en 1993 se comercializó Windows NT para estaciones de trabajo y servidores.



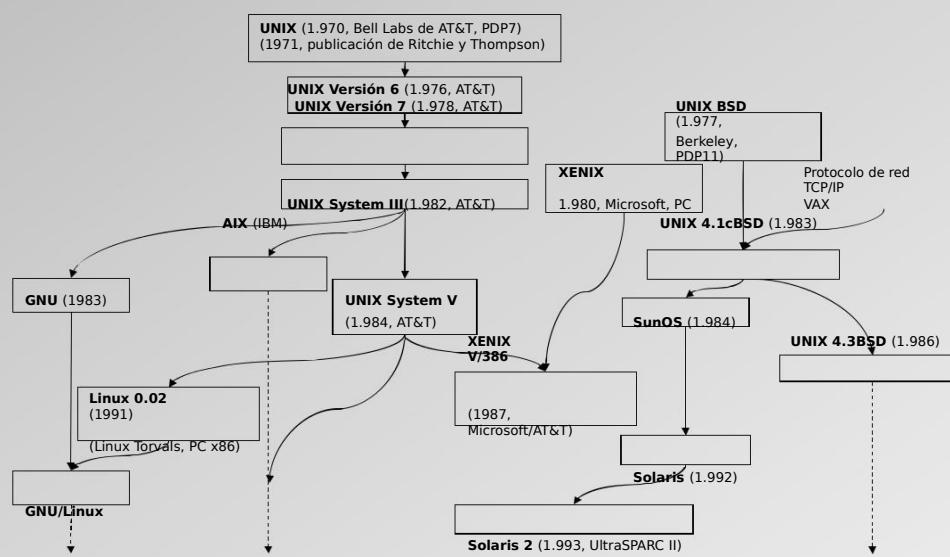
Evolución de los SO

- **UNIX:**

- Fue comercializado a comienzos de los 90
- Características:
 - Constituye un conjunto de familias de SO que comparten criterios de diseño e interoperatividad.
 - Se han desarrollado más de 100 SO a lo largo de 20 años.
 - Puede funcionar en muchos tipos de ordenadores.
 - Es la columna vertebral de Internet
 - Es un SO multiprogramación, multiusuario y multiprocesamiento
 - Es más difícil de aprender que Windows o Mac-OS



Evolución de los SO UNIX



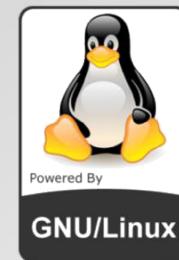
Evolución de los SO

- **Linux (1991):**

- lo creó un estudiante de la universidad de Helsinki.
- Fue concebido como una versión de Minix (parecido a UNIX) para PC compatible.
- Es un SO parecido al UNIX que funciona en desde supercomputadores a pequeños sistemas de electrónica de consumo.
- Es de código abierto
- 0.5-2% de todos los PC.

- **GNU-Linux**

- Sistema operativo de software libre, que contiene el núcleo de Linux



Evolución de los SO

- **Google Chrome OS**

- Está basado en el núcleo de Linux
- Proyectado para los usuarios que dedican gran cantidad de tiempo a Internet.
- Técnicamente es un navegador web.
- Se basa en la utilización de aplicaciones de Internet usadas a través de web para la realización de tareas tales como procesamiento de textos y visualización multimedia.



- **MacOs (80):**

- Desarrollado por Apple.
- SO de los Macintosh.
- Precursor de las GUI (ventanas, ratón, etc.).



- **iOS**

- Hasta Junio 2010 era conocido como iPhone OS
- Es el sistema operativo móvil de Apple
- Se desarrolló originalmente para iPhone, pero se sigue usando para el iPod Touch, iPad and Apple TV.

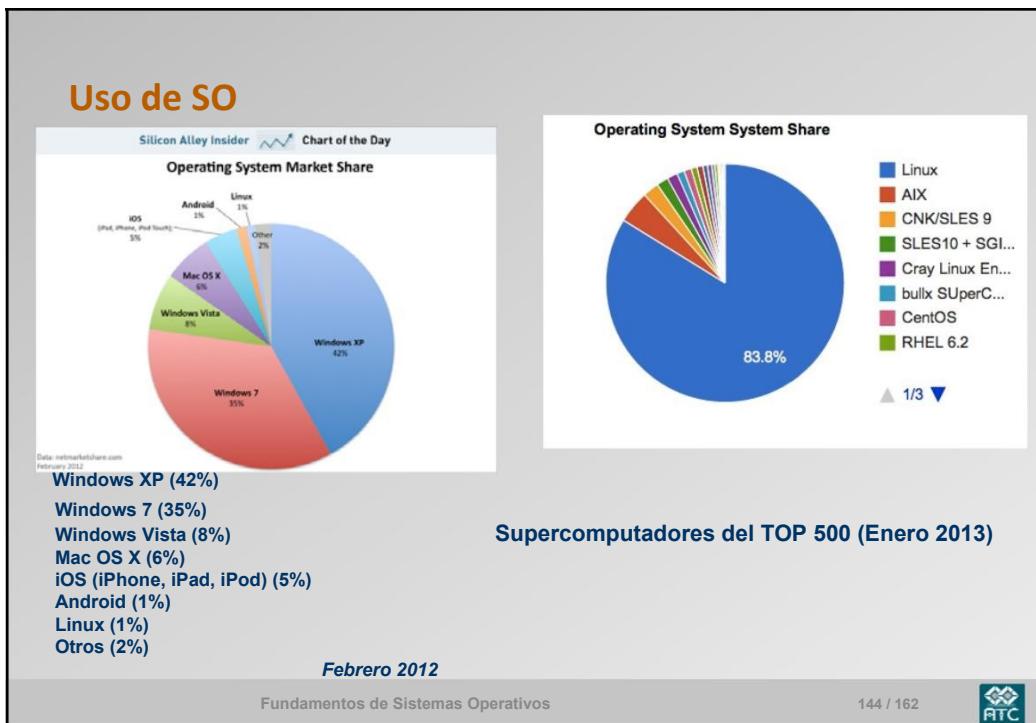
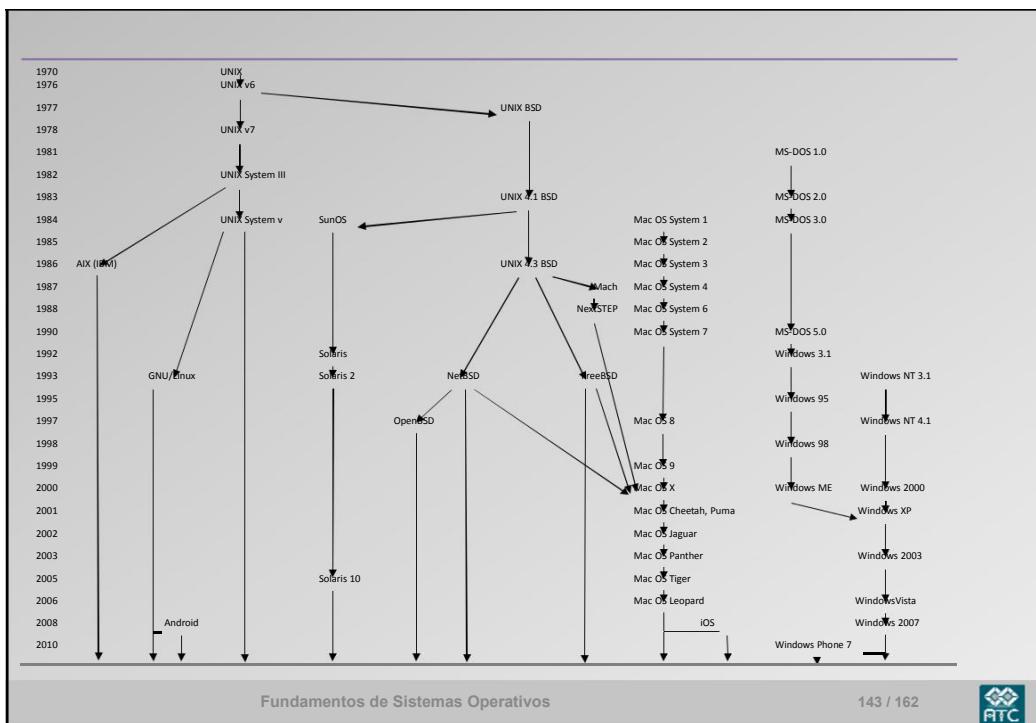


Evolución de los SO

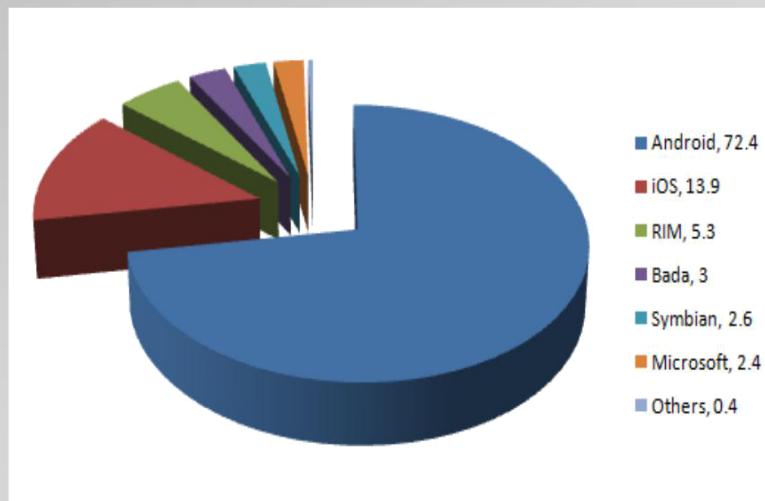
- **Otros SO:**

- IBM: z/OS, z/VM, y OS/2 (desarrollado conjuntamente con Microsoft)





SO para móviles Q3 - 2012



Fundamentos de Sistemas Operativos

145 /
120 



Fundamentos de Sistemas Operativos

146 /
120 

Contenidos

- **Definición**

- Objetivos y funciones
- Contexto
- Arranque de un computador

- **Gestión de recursos:**

- Gestión del procesador,
- Gestión de la memoria
- Gestión de E/S
- Gestión de archivos

- **Sistemas operativos más conocidos**

- **Seguridad**



>> W.Stalling, Sistemas Operativos, 5^a Ed.,
Prentice Hall, 2005; Cáp. 16, pp. 689-741

Orígenes de las amenazas a la seguridad

- **El origen de las amenazas a la seguridad de sistemas informáticos proviene fundamentalmente de:**

- la capacidad de compartir recursos
- el desarrollo de la conectividad de computadores, y fundamentalmente con el surgimiento de Internet ha dado lugar a la posibilidad de acceso físico desde cualquier computador a cualquier otro equipo informático.
- el desarrollo de programas malignos que atentan contra la integridad del software (cargables a través de soportes intercambiables o de Internet)

Tipos de amenazas

- **Interrupción:**

– se destruye un elemento físico del sistema y no puede continuar el flujo de información, o un elemento software imposibilitando el uso adecuado del mismo. Atacando a la disponibilidad.

- **Interceptación:**

– un ente no autorizado consigue acceder a un elemento. Se priva del secreto.

- **Modificación:**

– una parte no autorizada consigue acceder y falsificar algún elemento, alterando la integridad.

- **Invención:**

– consiste en la inserción de objetos falsos en el sistema, dañando la autenticidad.

Tipos de amenazas

Amenazas a la seguridad y elementos [STA01]			
	Disponibilidad	Secreto	Integridad/Autenticidad
Hardware	Robo o inutilización de equipos, privación del servicio		
Software	Eliminación de programas, denegar el acceso a los usuarios	Realización de copias no autorizadas de software	Alteración de un programa en funcionamiento, haciéndolo fallar durante la ejecución o haciendo que realice alguna tarea no pretendida
Datos	Eliminación de archivos, denegar el acceso a los usuarios	Lecturas de datos no autorizadas. Un análisis de datos estadísticos revela datos ocultos	Cambio en archivos existentes o invención de nuevos archivos
Redes y Líneas de comunicación	Destrucción o eliminación de mensajes. Las líneas de comunicación o redes no están disponibles	Lectura de mensajes. Observación del tráfico de mensajes.	Mensajes modificados, retardados, reordenados o duplicados. Invención de mensajes falsos.

Tipos de amenazas

- **Naturaleza de las amenazas:**

- Amenazas pasivas: se suelen denominar escuchas

- Divulgación del contenido del mensaje
 - Análisis del tráfico

- Amenazas activas:

- implican una modificación del flujo de datos o una creación de flujos falsos.

- Suplantación
 - Repetición
 - Modificación del contenido de los mensajes
 - Privación de servicios

Software maligno

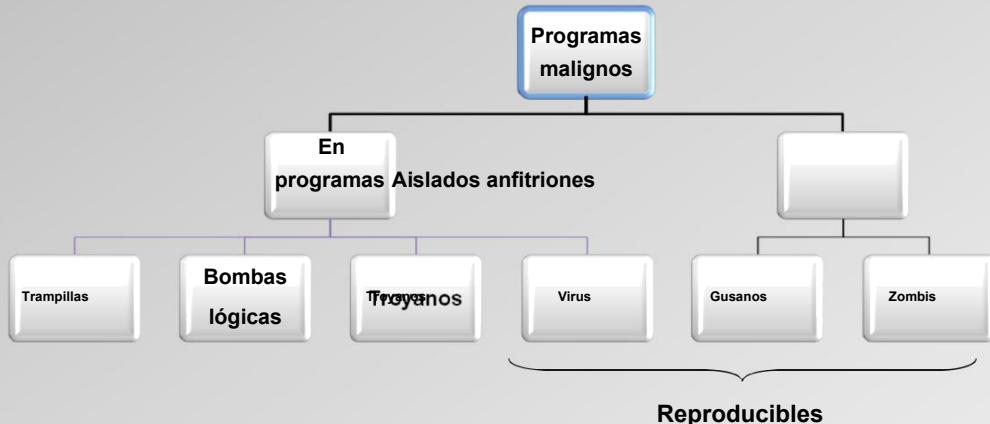
- El software o programas malignos son programas que se aprovechan de los puntos vulnerables de los sistemas informáticos.

- Suelen ir camuflados dentro de programas legítimos (con frecuencia de libre distribución) o en suplantaciones de éstos.



Software maligno

- Clasificación de los programas malignos



Software maligno

- Trampilla:

– es un punto de entrada secreto a un programa que permite que quien la conoce pueda acceder sin pasar por los procedimientos usuales de seguridad de acceso. Es muy difícil implementar controles para trampillas en el sistema operativo. Las medidas de seguridad deben centrarse en el desarrollo de programas y en sus actualizaciones, cuyos puntos de entrada estén controlados.

- Bomba lógica:

– es un trozo de código insertado en alguna parte de un programa y que produce el daño cuando se cumplen ciertas condiciones.

- Caballo de Troya:

– es un programa o procedimiento que contiene un código oculto que, cuando se invoca, realiza alguna función dañina o no deseada. También se pueden usar para efectuar operaciones indirectas que un usuario no podría realizar directamente. También pueden pretender la destrucción de datos.

Software maligno

- **Gusanos:**

- emplean las conexiones de la red para propagarse de un sistema a otro. Una vez que se activa el gusano puede comportarse como un virus o implantar un caballo de Troya o realizar cualquier acción destructora. Para poder reproducirse necesitan algún tipo de vehículo como puede ser el correo electrónico, la capacidad de ejecución remota de un sistema o la capacidad de conexión remota.

- **Zombie:**

- es un programa que toma posesión de otro computador conectado a Internet y lo usa para lanzar ataques desde él a terceros computadores, de forma que es difícil detectar al creador. Normalmente se usan para atacar a un sitio Web.

Software maligno

- **Virus:**

- es un programa que puede “infectar” a otros programas modificándolos. Un virus porta en su código la receta para hacer copias perfectas de sí mismo. También puede propagarse de un computador a otro a través de discos o al enviar programas infectados por la red.

- Un virus pasa por cuatro etapas:

- Fase latente: el virus está inactivo hasta que algún suceso lo active.
- Fase de propagación: el virus se copia, de forma que cada programa infectado contiene ahora una copia del virus.
- Fase de activación: el virus se activa por algún suceso para llevar a cabo la función para la que se creó.
- Fase de ejecución: lleva a cabo su función, que puede ser dañina o no.



Software maligno

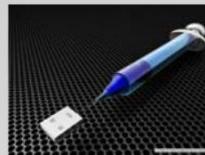
- A medida que se han ido desarrollando antivirus se han ido creando nuevos tipos de virus:
 - Virus parásitos: van junto a archivos ejecutables.
 - Virus residentes en memoria: se alojan en memoria principal como parte de un programa residente, infectando a todos los programas que se ejecutan.
 - Virus residentes en el sector de arranque: infectan el sector principal de arranque y se propagan cuando el sistema arranca desde el disco que contiene el virus.
 - Virus clandestino: un virus que se esconde de la detección con un software antivirus.



Software maligno

- Virus polimorfo: un virus que muta con cada infección, creando copias que son funcionalmente equivalentes pero con distintos patrones de bits.
- Virus de macro: una macro se puede activar cuando se de una determinada secuencia de teclas o un suceso determinado, de forma que cuando se ejecuta se activa el virus. Suelen estar insertados en documentos. Por ejemplo, en Word o Excel se pueden definir macros autoejecutables o no, para generar un formulario, en estos ejecutables se pueden insertar virus, que se activan con tan sólo abrir el documento.
- Virus de correo electrónico: son virus que se activan al abrir un fichero adjunto o al abrir el correo. En este último caso el virus utiliza el lenguaje Visual Basic para activarse.

Estrategias de los antivirus



- Detección
 - Identificación
 - Eliminación
-
- INTECO es el Centro de Referencia en Seguridad Informática a nivel nacional.
 - <http://www.inteco.es/Seguridad>
 - Centro de Respuesta a Incidentes de Seguridad (CERT)
 - Ofrece soluciones reactivas a incidentes informáticos, servicios de prevención frente a posibles amenazas y servicios de información, concienciación y formación en materia de seguridad.
 - Estadísticas:
http://cert.inteco.es/virusMoreDetSearch/Actualidad/Actualidad_Virus/Buscador_Virus_Detectados/?postAction=getVirusValues.
- EEUU: <http://www.us-cert.gov/>

Intrusos

- Intruso, pirata informático, hacker o cracker
 - Son una de las peores amenazas a la seguridad.
 - Existen tres tipos de intrusos:
 - Suplantador: es un individuo que no está autorizado para usar un computador y que entra en el sistema de control de accesos para autorizarse.
 - Abusador: es un usuario que puede entrar en el sistema pero que accede a recursos a los que no está autorizado.
 - Usuario clandestino: es una persona que está a cargo del control de supervisión del sistema y aprovecha este control para evadir la auditoria y el control de acceso.

Intrusos

- El objetivo de los intrusos es obtener acceso a un sistema o aumentar sus privilegios de accesibilidad al mismo. Para ello tiene que obtener una información protegida.
- Estrategias de protección:
 - Control de acceso
 - Identificación y autentificación
 - Cifrado
 - Cortafuegos



Intrusos

- **Sistemas de detección de intrusos:**
 - IDS: Intrusion Detection System, is software and/or hardware designed to detect unwanted attempts at accessing, manipulating, and/or disabling of computer systems, mainly through a network, such as the Internet.
 - NIDS: Network intrusion detection system, a system that tries to detect malicious activity by monitoring network traffic
 - HIDS: host-based intrusion detection system. Monitors and analyzes the internals of a computing system rather than the network packets on its external interfaces (as a NIDS haría).
- **Detección:**
 - Si se detecta un intruso, puede ser identificado y expulsado del sistema
 - Puede disuadir a los intrusos a intentar atacar el sistema

SO y seguridad

- **El SO debe proteger al computador de las amenazas a la seguridad.**
 - Memoria
 - Dispositivos de E/S (discos, impresoras, etc.)
 - Programas
 - Datos

SO y seguridad

- **Memoria**
 - Es fundamental en sistemas de multiprogramación
 - un proceso podría invadir la zona de memoria de otro
 - Si no hay segmentos o datos compartidos, el SO debe asegurar que se accede a ellos únicamente por el proceso a los que estén asignados.
 - Si se comparten segmentos o datos: una tabla para cada proceso que indique a los que puede acceder.
 - Si hay segmentación o segmentación paginada, es relativamente fácil de gestionar.
 - Ejemplo:
 - IBM z/OS: para cada marco de página (en la tabla) se incluyen 5 bits de protección (1 para tipo de acceso, y 4 de clave de acceso)

SO y seguridad

- **Control de acceso orientado al usuario:**

- Servicio de conexión de usuario:
 - Identificación del usuario (ID)
 - Contraseña (*password*) ↗ Muy frecuentemente mal asignadas
- Mejora de seguridad: control de acceso de dos niveles:
 - El propio computador
 - Control de acceso a la red de datos (ejemplo VPN)

SO y seguridad

- **Control de acceso orientado a los datos**

- En general, no es suficiente un control de acceso de usuario, ya que dependiendo de quién sea puede tener acceso a distintos programas o datos, con distintos privilegios.
- A cada usuario se le asocia un perfil que especifica las operaciones permitidas en los accesos a archivos o programas.
- El SO puede aplicar reglas de acceso basadas en el perfil del usuario para autorizar los accesos.
- Sistemas de Gestión de Bases de Datos: deben controlar el acceso a determinados registros o a campos de los mismos.
- El control de acceso puede hacerse con una tabla que incluya:
 - Sujeto: elemento (proceso) capaz de acceder a los programas, elementos de una BD, etc.
 - Objeto: Todo elemento al que se debe acceder de forma controlada (programa, archivo, segmentos de memoria, registros de BD, etc.)
 - Derechos de acceso: forma de acceso (leer, escribir, ejecutar)

Seguridad en Windows

- **Servicio de control de acceso uniforme**

- Se aplica a:

- Procesos
 - Hebras
 - Archivos
 - Ventanas, etc.

Seguridad en Windows

- **Concepto:**

- Entidades de seguridad:

- **Testigo de acceso** ↗ asociado a cada proceso
 - El SO, al autenticar al usuario, le asigna un proceso de seguridad con un testigo de acceso, que contiene un identificador de seguridad (SID), con el que es reconocido el usuario a efectos de seguridad
 - **Descriptor de seguridad** ↗ asociado a cada objeto de acceso
 - Contiene una lista de control de accesos que especifica los derechos para cada usuario o grupo de usuarios a dicho objeto.
 - Cuando un proceso accede a un objeto, se coteja el testigo de acceso del proceso con el descriptor de seguridad del objeto.



"Grandma, have you ever thought of just upgrading Grandpa's operating system?"