



Tema 1. Introducción

Índice

- Concepto de Sistema Operativo (SO)
- Modo supervisor vs. Modo usuario
- Llamadas al sistema
- Interrupciones
- Arranque del SO
- Activación del SO
- Estructura del SO
- El intérprete de comandos
- Taxonomía de los SO

Tema 1. Introducción

Bibliografía

- J. Carretero et al. *Sistemas Operativos: Una Visión Aplicada*. McGraw-Hill. 2007. Temas 1 y 2

Tema 1. Introducción


Resultados de aprendizaje

- Describir la función del SO y su interacción con la arquitectura del computador y las aplicaciones que se ejecutan sobre él.
 - Definir qué es un sistema operativo y sus funciones
 - Describir qué es una llamada al sistema y cómo se realizan
 - Describir la relación entre llamadas al sistema, interrupciones y SO
 - Describir cómo y cuándo se activa el sistema operativo
 - Describir qué ocurre en el computador cuando se activa el sistema operativo
 - Describir el proceso de arranque del computador
 - Describir qué es un intérprete de órdenes
 - Describir los componentes de un SO
 - Describir los tipos de SO en base a su estructura



Tema 1. Introducción

Índice

-  ■ Concepto de Sistema Operativo (SO)
- Modo supervisor vs. Modo usuario
- Llamadas al sistema
- Interrupciones
- Arranque del SO
- Activación del SO
- Estructura del SO
- El intérprete de comandos
- Taxonomía de los SO



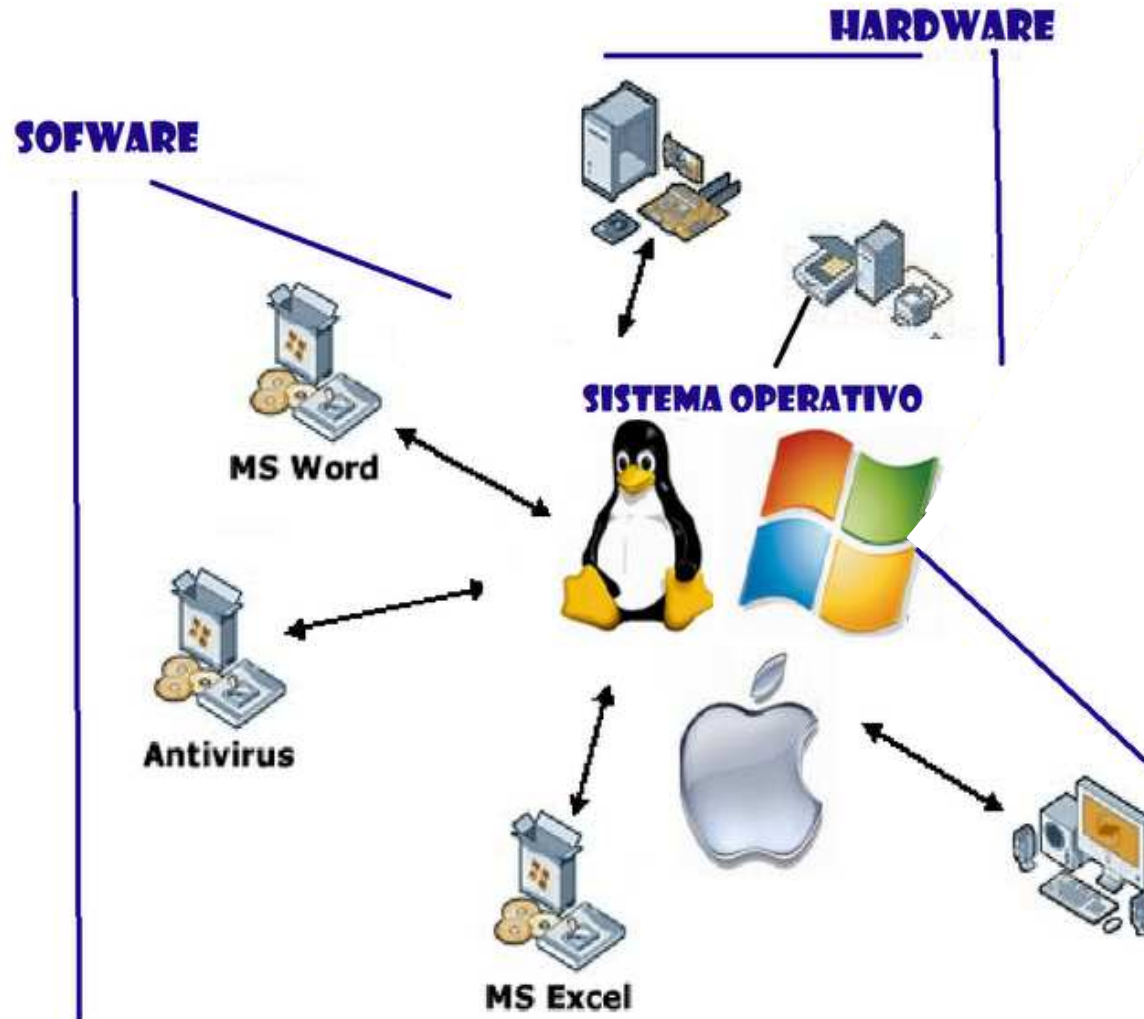
Concepto de SO

- Al inicio de este video se describe qué es un SO:

<http://www.youtube.com/watch?v=wn3vLhfGTJA&feature=related>

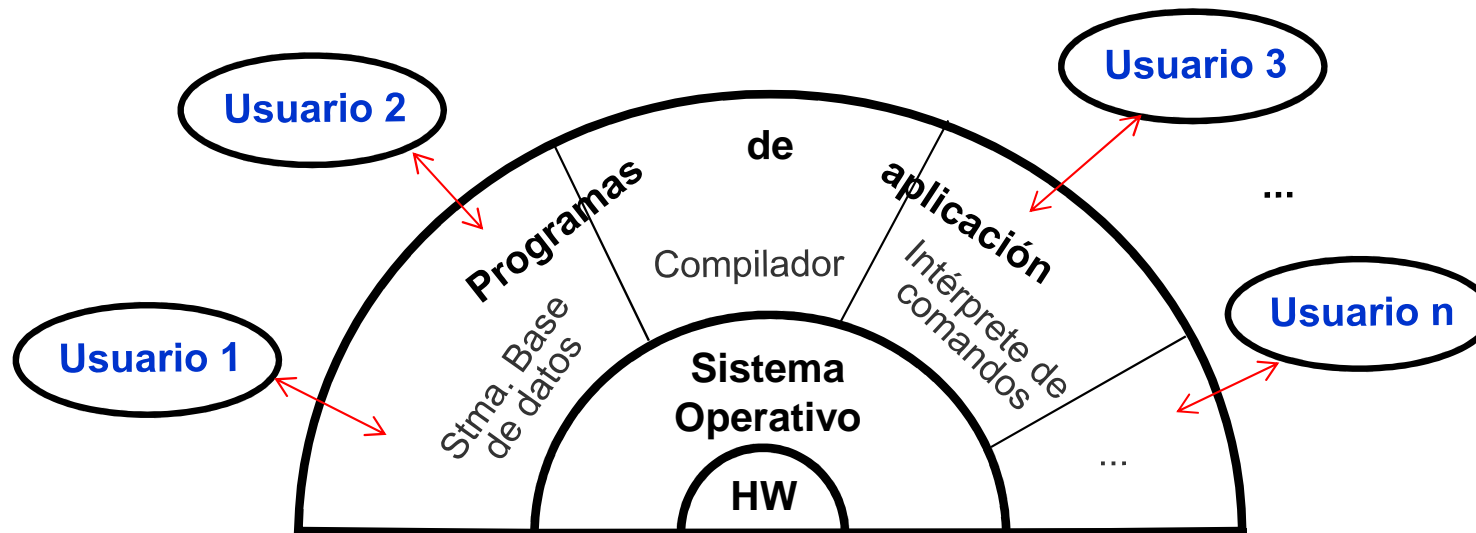
¡Ojo! Hacia la mitad del video hay una errata

Concepto de SO



Concepto de SO

- Perspectiva abstracta de un Sistema de Información:





Concepto de SO

- **Perspectiva abstracta de un Sistema de Información (cont.):**
 - ◆ Hardware (HW):
 - Proporciona los recursos básicos del sistema (CPU, memoria, dispositivos de entrada/salida -E/S-, etc.).
 - ◆ Sistema operativo (SO):
 - Controla y coordina el uso del HW entre los diversos programas de aplicación de los diferentes usuarios.
 - ◆ Programas de aplicación:
 - Definen la forma en que se usan los recursos del sistema para resolver los problemas de computación de los usuarios.
 - Demandan servicios del SO mediante llamadas al sistema.
 - ◆ Usuarios:
 - Entes que hacen uso del computador (personas y otras computadoras y máquinas).



Concepto de SO

- **Definición de SO:**
 - ◆ **SO como interfaz usuario-computador (o máquina extendida):**
 - Programa que actúa de intermediario entre los programas de aplicación de un computador y el HW
 - ◆ **SO como gestor de recursos:**
 - Entidad que gestiona los recursos HW y software (SW) entre los procesos y usuarios que compiten por ellos



Concepto de SO

■ Objetivos del SO:

- ◆ Proporcionar un entorno en el que el usuario pueda ejecutar sus programas
 - Comodidad para el usuario
 - Ofrecer una abstracción de los recursos HW para que el SW de aplicación no necesite conocer las peculiaridades de cada máquina
- ◆ Gestionar los recursos HW y SW
 - Decidir quién, cuándo y cuánto tiempo se usa un recurso
 - Resolver conflictos entre demandas concurrentes de recursos
 - Utilización eficiente o eficaz de recursos



Concepto de SO

- **Comunicación entre procesos de usuario y el SO:**
 - ◆ Las **llamadas al sistema** son la puerta de acceso a los servicios ofrecidos por el SO

- **Comunicación dentro del SO:**
 - ◆ Modelo cliente-servidor: Comunicación por mensajes
 - ◆ Modelo biblioteca: Llamadas a procedimientos

Arquitectura HW del sistema

Asumiremos ...


- **Modelo básico del sistema:**

- ◆ Modo usuario vs. Modo supervisor.
- ◆ Comunicación CPU-E/S basada en interrupciones.
- ◆ Acceso al SO mediante llamadas al sistema.



Tema 1. Introducción

Índice

- Concepto de Sistema Operativo (SO)
-  ■ Modo supervisor vs. Modo usuario
- Llamadas al sistema
- Interrupciones
- Arranque del SO
- Activación del SO
- Estructura del SO
- El intérprete de comandos
- Taxonomía de los SO



Modo usuario vs. Modo supervisor

- En un SO multiprogramado la compartición de recursos del sistema requiere:
 - ◆ Protección del SO.
 - ◆ Protección de los demás programas y datos.
- Solución: Apoyo del HW para permitir dos modos de ejecución:
 - ◆ *Modo usuario* (modo restringido):
 - Ejecución de instrucciones del código del usuario.
 - Acceso a direcciones del programa y de datos del usuario.
 - Ejecución de instrucciones no privilegiadas.
 - ◆ *Modo supervisor* (modo privilegiado, no restringido o monitor):
 - Ejecución de instrucciones del código del SO.
 - Acceso a direcciones del SO y de usuarios.
 - Ejecución de instrucciones privilegiadas y no privilegiadas.

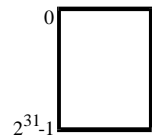


Modo usuario vs. Modo supervisor

D0			
D1			
D2			
D3			
D4			
D5			
D6			
D7			

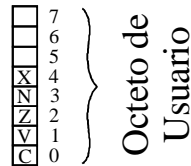
A0		
A1		
A2		
A3		
A4		
A5		
A6		
A7		

--



Mapa de memoria

Registro de estado

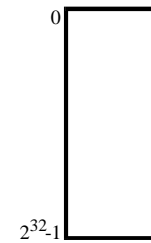


Modo usuario

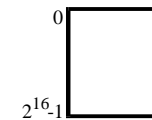
D0			
D1			
D2			
D3			
D4			
D5			
D6			
D7			

A0		
A1		
A2		
A3		
A4		
A5		
A6		
A7		
A7'		

--



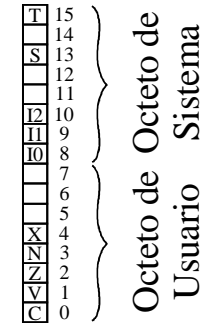
Mapa de memoria



Mapa de E/S

Modo supervisor

Registro de estado



Juego de Instrucciones




Modo usuario vs. Modo supervisor

- **Cambios de un modo de ejecución a otro:**
 - ◆ El HW cambia a modo supervisor cuando ocurre un error o cuando se desea realizar una operación de E/S.
 - ◆ Antes de que el SO obtenga el control del computador el sistema cambia a modo supervisor.
 - ◆ Antes de pasar el control al programa de usuario, el sistema cambia a modo usuario

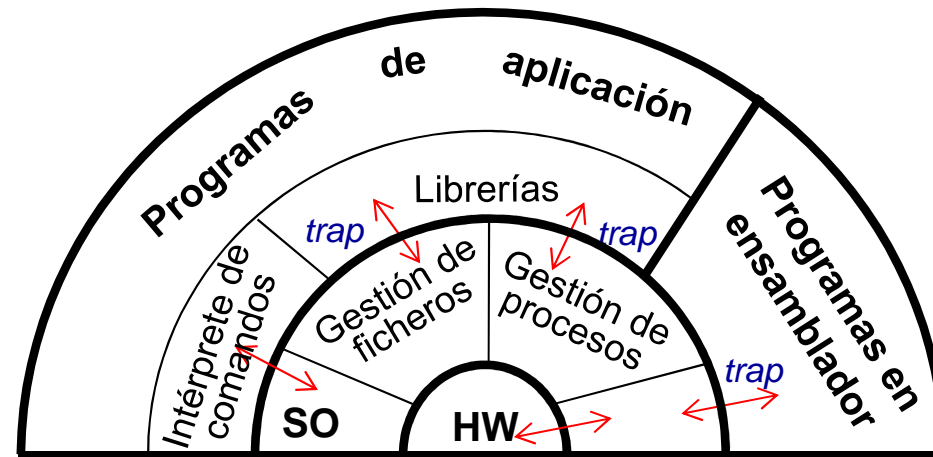


Tema 1. Introducción

Índice

- Concepto de Sistema Operativo (SO)
- Modo supervisor vs. Modo usuario
-  ■ Llamadas al sistema
- Interrupciones
- Arranque del SO
- Activación del SO
- Estructura del SO
- El intérprete de comandos
- Taxonomía de los SO

Llamadas al sistema



■ ¿Qué son las llamadas al sistema?

- ◆ Puerta de acceso a los servicios ofrecidos por el SO.
- ◆ Invocadas por procesos en modo usuario.
- ◆ Ejecutadas en modo supervisor.



Llamadas al sistema

- **Interfaz ofrecida por las llamadas al sistema:**
 - ◆ Interfaz ofrecida por el núcleo del SO:
 - Definida a nivel de lenguaje ensamblador.
 - Depende del HW sobre el que se está ejecutando el SO.
 - ◆ Interfaz ofrecida al programador (usuario):
 - Funciones dentro de una librería (biblioteca) de usuario.

Llamadas al sistema

■ Ejecución de una llamada al sistema:

- ◆ Introducir los parámetros de la llamada (en registros o en pila).
- ◆ Generar una interrupción software (trap).
- ◆ Pasar de modo usuario a modo supervisor y cambiar de pila usuario a pila supervisor.
- ◆ Guardar todos los registros del procesador.
- ◆ Saltar a la función que implementa el servicio pedido.
- ◆ Restaurar los registros del procesador.
- ◆ Reponer el modo del procesador y la pila de usuario.
- ◆ Retornar de la interrupción.
- ◆ Devolución de parámetros.





Llamadas al sistema


■ Clasificación de llamadas al sistema:

- ◆ Control de procesos (*fork, execvp, wait, exit, getpid, ...*).
- ◆ Manipulación de ficheros (*pipe, dup, creat, open, lseek, stat, mkdir, ...*).
- ◆ Manipulación de dispositivos.
- ◆ Monitorización del sistema.
- ◆ Comunicaciones.



Tema 1. Introducción

Índice

- Concepto de Sistema Operativo (SO)
- Modo supervisor vs. Modo usuario
- Llamadas al sistema
-  ■ Interrupciones
- Arranque del SO
- Activación del SO
- Estructura del SO
- El intérprete de comandos
- Taxonomía de los SO



Interrupciones

- **Secuencia de trabajo del computador:** Se descompone en 5 fases de ejecución que se repite de forma infinita:
 - ◆ F0: Lectura de instrucción apuntada por CP e incremento de CP
 - ◆ F1: Decodificación
 - ◆ F2: Lectura de operandos
 - ◆ F3: Ejecución de la operación en la ALU
 - ◆ F4: Almacenamiento del resultado

- **Esa secuencia se rompe con:**
 - ◆ Instrucciones de salto. Su ejecución modifica el CP. Forman parte del código del programa.
 - ◆ Interrupciones. El procesador automodifica el CP.

Interrupciones

- **Interrupción:** Mecanismo que permite romper (interumpir) la secuencia de ejecución de un procesador





Interrupciones

■ Clasificación de interrupciones:

- ◆ Interrupciones HW (o simplemente interrupciones):
 - Permiten comunicación entre el HW y el SO.
 - Causadas por eventos externos al proceso (operaciones E/S, reloj, ...).
 - Se producen en cualquier momento y se atienden entre la ejecución de dos instrucciones.
- ◆ Interrupciones SW (excepciones o trap)
 - Causadas por el propio proceso ante un evento inesperado o indeseado (división por cero, error de direccionamiento, fallo de página, etc.) o ante una petición de usuario (llamada al sistema).
 - Se producen durante la ejecución de una instrucción.



Interrupciones

- **Otra clasificación de interrupciones:**
 - ◆ Interrupciones HW:
 - Externas (operaciones E/S, reloj, ...).
 - Internas (división por cero, error de direccionamiento, fallo de página, etc.) .
 - ◆ Interrupciones SW:
 - Causadas ante una petición de usuario (llamada al sistema)

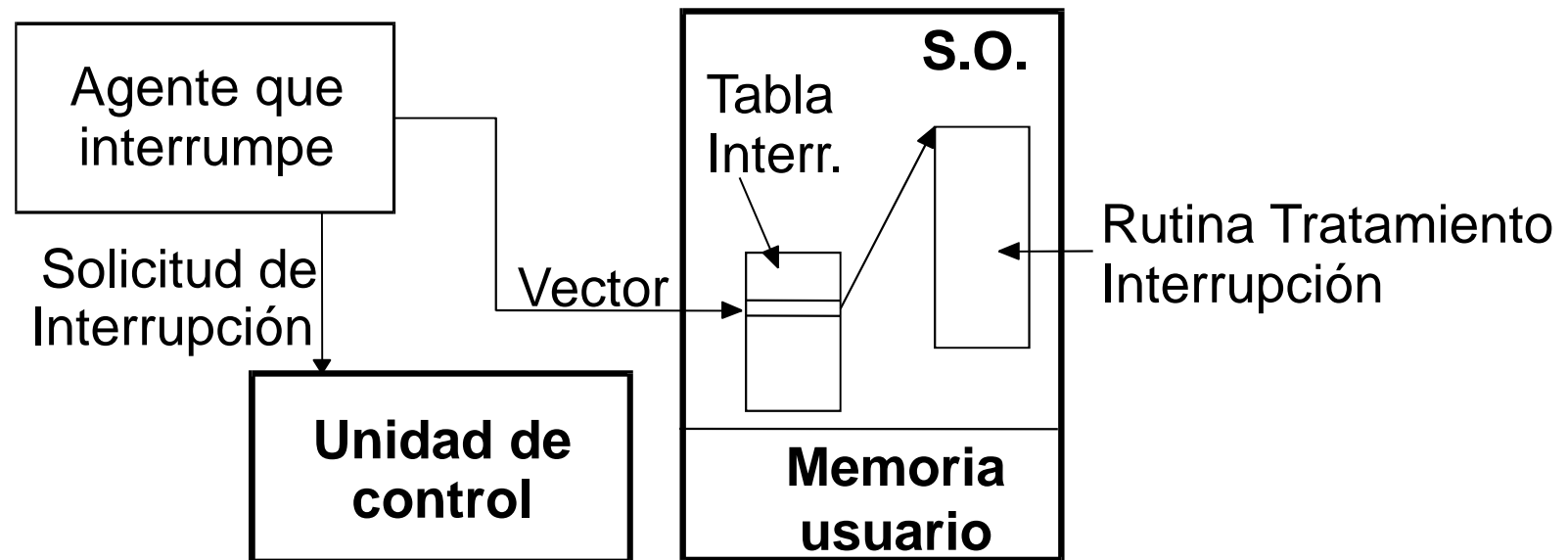


Interrupciones

- Un SO está dirigido por interrupciones.
- Gestión de interrupciones: muy dependiente de la arquitectura del computador.
- **Tratamiento de una interrupción:**
 - ◆ Pasar a modo supervisor.
Todas las interrupciones se atienden en modo supervisor
 - ◆ Determinar la causa (número) de la interrupción.
 - ◆ Buscar el vector de interrupción (contiene la dirección de la rutina de tratamiento de la interrupción).
 - ◆ Salvar (apilar) contexto del proceso actual (dirección de instrucción interrumpida, registros de CPU, etc.).
 - ◆ Ejecutar la rutina de tratamiento de la interrupción.
 - ◆ Recuperar (desapilar) el contexto previo del proceso.
 - ◆ **Volver al modo de ejecución anterior**

Interrupciones

- Tratamiento de una interrupción (cont.):





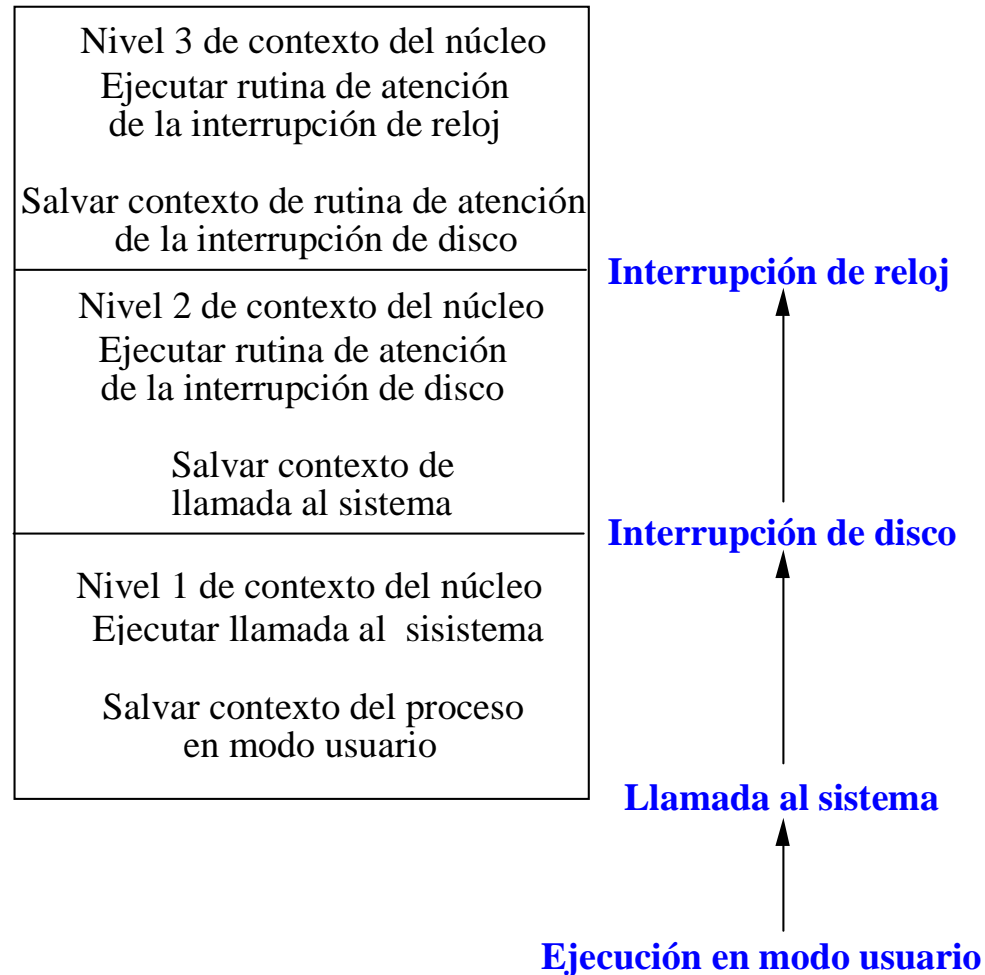
Interrupciones

■ Anidamiento de interrupciones:

- ◆ Normalmente, al procesar una interrupción se deshabilitan las restantes interrupciones para evitar pérdidas de interrupciones.
- ◆ En arquitecturas con gestión compleja de interrupciones:
 - Se asignan prioridades a las interrupciones según su importancia relativa
 - Sólo se deshabilitan las interrupciones de prioridad inferior a la que se está atendiendo

Interrupciones


■ Anidamiento de interrupciones (cont.):





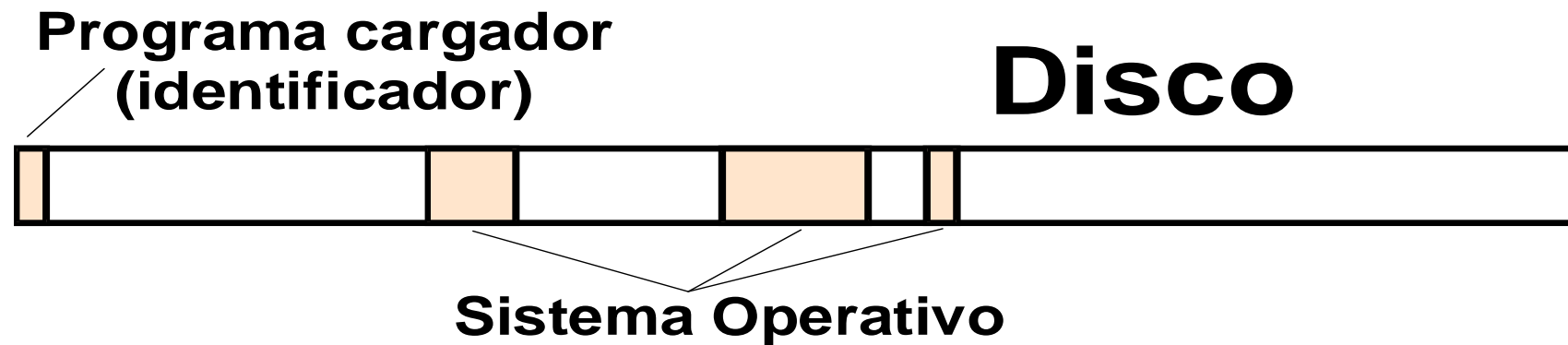
Tema 1. Introducción

Índice

- Concepto de Sistema Operativo (SO)
- Modo supervisor vs. Modo usuario
- Llamadas al sistema
- Interrupciones
-  ■ Arranque del SO
- Activación del SO
- Estructura del SO
- El intérprete de comandos
- Taxonomía de los SO

Arranque del SO

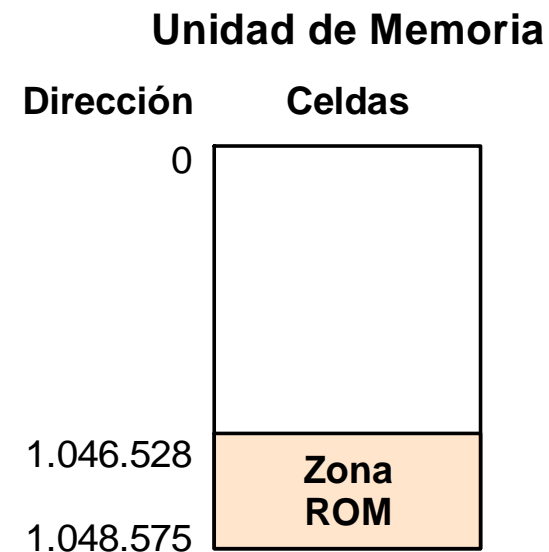
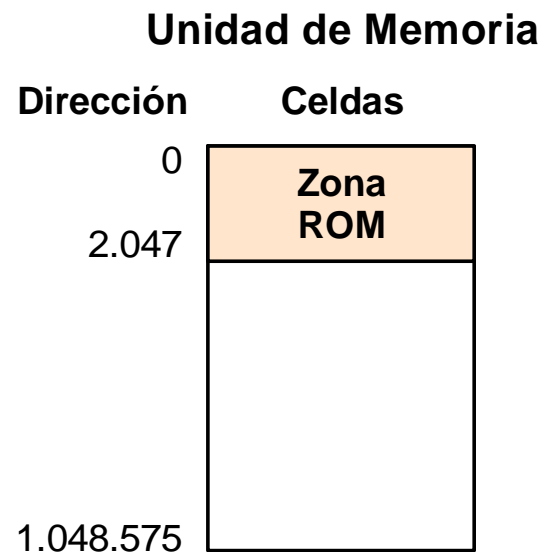
- La computadora necesita siempre un programa en MC para hacer una actividad
- El SO está almacenado en disco



Arranque del SO

■ Iniciador ROM:

- ◆ El arranque o el reset carga en CP la dirección de arranque del cargador ROM
- ◆ En el caso del PC la ROM contiene, además, SW de E/S (BIOS)





Arranque del SO

- Se ejecuta el **cargador ROM** del sistema:
 - ◆ Test del sistema
 - ◆ Trae a memoria el cargador (boot) del SO
 - ◆ Le da control


También puede ser un gestor de arranque como GRUB o LILO



Arranque del SO

- El boot del SO carga el SO residente y da control al programa de **arranque del SO**, que:
 - ◆ Comprueba el HW
 - ◆ Comprueba el sistema de ficheros
 - ◆ Establece las tablas del SO
 - ◆ Crea procesos (según el tipo de SO)
 - Proceso INIT
 - Procesos o tareas del sistema (demonios)
 - Procesos de login (uno por terminal)
 - ◆ Tras la autenticación, el proceso login se convierte en shell

Índice

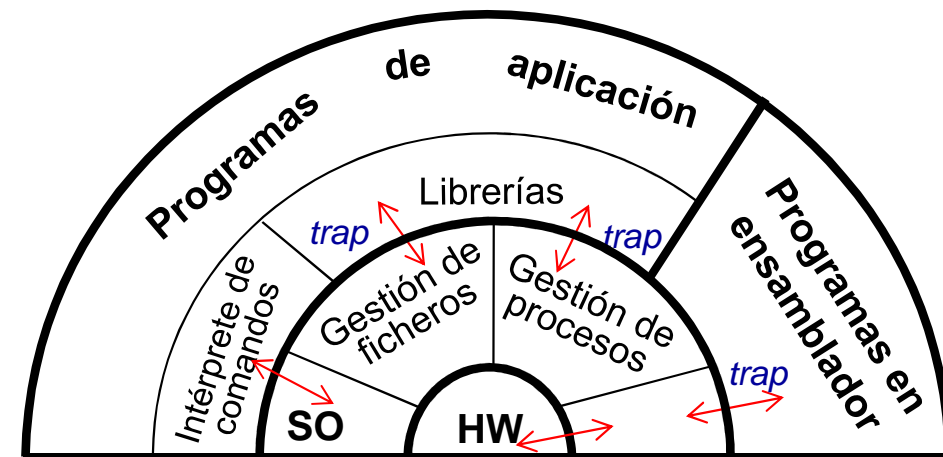
- Concepto de Sistema Operativo (SO)
- Modo supervisor vs. Modo usuario
- Llamadas al sistema
- Interrupciones
- Arranque del SO
-  ■ Activación del SO
- Estructura del SO
- El intérprete de comandos
- Taxonomía de los SO

Activación del SO

■ Fuentes de las peticiones:

- ◆ Llamadas al sistema emitidas por los programas
- ◆ Interrupciones producidas por los periféricos
- ◆ Condiciones de excepción de programa

Pero en todos los casos la activación del SO se realiza mediante el mecanismo de interrupciones





Activación del SO





Activación del SO vía llamada al sistema

- Una hipotética implementación de la llamada *fork*:

Mirar 1ª edición del libro de Carretero et al.

Función de librería ≠ Llamada al sistema

```
int fork() {
    int r;
    LOAD R8, FORK_SYSTEM_CALL
    TRAP
    LOAD r, R9
    return r;
}
```

→ Paso de parámetros

→ Invocación del servicio fork al SO

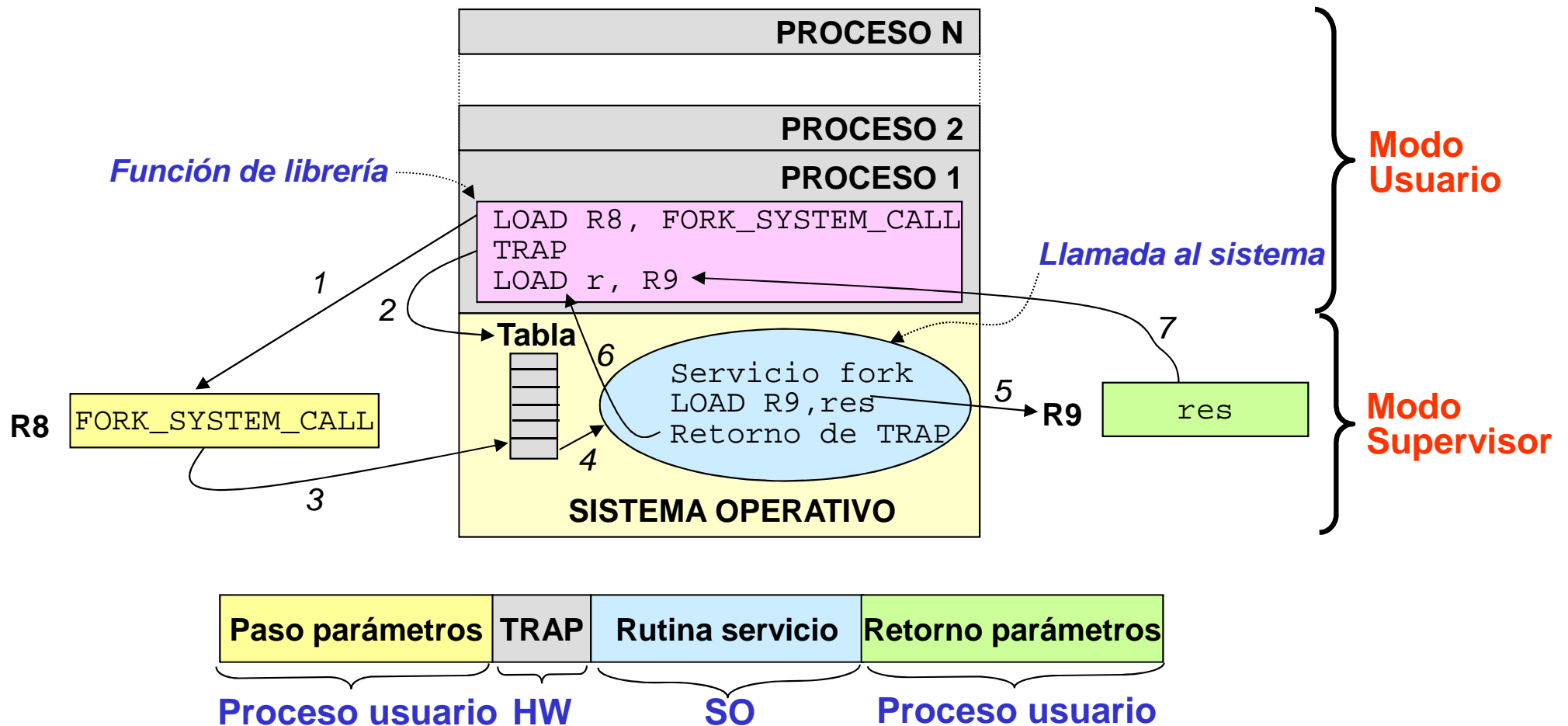
→ Retorno de parámetros

Llamada al sistema



Activación del SO vía llamada al sistema


- Una hipotética implementación de la llamada *fork* (cont.):
 - ◆ Pasos de la llamada al sistema:





Tema 1. Introducción

Índice

- Concepto de Sistema Operativo (SO)
- Modo supervisor vs. Modo usuario
- Llamadas al sistema
- Interrupciones
- Arranque del SO
- Activación del SO
-  ■ Estructura del SO
- El intérprete de comandos
- Taxonomía de los SO



Estructura del SO

- **Componentes típicos de un SO:**
 - ◆ Gestor de procesos
 - ◆ Gestor de memoria principal
 - ◆ Gestor de almacenamiento secundario
 - ◆ Gestor de ficheros
 - ◆ Gestor del sistema de E/S
 - ◆ Sistema de protección
 - ◆ Gestión de red

<http://www.youtube.com/watch?v=863IxBtarhZE&feature=related>



Estructura del SO

■ Gestión de procesos:

- ◆ Concepto de proceso:
 - Programa en ejecución.
 - Necesita ciertos recursos para realizar su tarea: tiempo de CPU, memoria, ficheros y dispositivos de E/S.
- ◆ Funciones del SO respecto a la gestión de procesos:
 - Crear y eliminar procesos.
 - Suspende y reanuda la ejecución de procesos.
 - Proporcionar mecanismos para:
 - Sincronización de procesos.
 - Comunicación de procesos.
 - Prevención de bloqueos.



Estructura del SO

■ Gestión de memoria principal:

◆ Concepto de memoria:

- Sucesión de celdas (bytes, palabras, ...) cada una con su dirección.
- Almacenamiento de datos e instrucciones de un proceso.
- Depósito de datos de acceso rápido entre CPU y dispositivos de E/S.
- Tamaño reducido.
- Información volátil.

◆ Funciones del SO respecto a la gestión de memoria:

- Conocer el estado de las celdas: libres u ocupadas.
- Decidir qué procesos se cargan en memoria.
- Asignar y recuperar el espacio en memoria.



Estructura del SO

- **Gestión de almacenamiento secundario:**
 - ◆ Almacenamiento de grandes cantidades de información no volátil (discos magnéticos, discos ópticos, ...).
 - ◆ Respaldo a la memoria principal.
 - ◆ Funciones del SO respecto a la gestión de almacenamiento secundario:
 - Gestión del espacio libre.
 - Asignación y liberación del espacio.
 - Planificación de las peticiones de disco.



Estructura del SO

■ Gestión de ficheros:

◆ Concepto de fichero:

- Conjunto de información relacionada definida por su creador.
- Unidad de almacenamiento lógico de información.

◆ Los ficheros se organizan en directorios para facilitar su uso.

◆ Funciones del SO respecto a la gestión de ficheros:

- Creación y borrado de ficheros y directorios.
- Primitivas para manipular ficheros y directorios.
- Mapeado de ficheros entre memoria y almacenamiento secundario.
- Copias de seguridad de ficheros.



Estructura del SO

- **Gestión de dispositivos de E/S:**
 - ◆ Ocultación al usuario de las particularidades HW de los dispositivos de E/S.
 - ◆ Un sistema de E/S ejemplo podría consistir en:
 - Sistema de memoria caché como buffer.
 - Interfaz general con los dispositivos.
 - Manejadores para dispositivos HW específicos.
 - ◆ Sólo el manejador de dispositivo conoce las particularidades de dicho dispositivo.



Estructura del SO

■ Sistema de protección :

- ◆ Necesidad de un mecanismo para controlar el acceso de programas, procesos o usuarios a los recursos del usuario y del sistema.
- ◆ Funciones del SO respecto al sistema de protección:
 - Determinar si el acceso está o no autorizado.
 - Determinar los controles a realizar.




Estructura del SO

- **Gestión de red (sistemas distribuidos):**
 - ◆ Concepto de sistema distribuido:
 - Conjunto de procesadores, cada uno con su propio reloj y memoria local.
 - Interconectados por una red de comunicaciones.
 - Compartición de recursos entre nodos interconectados.
 - ◆ El acceso a recursos compartidos permite:
 - Aumentar la velocidad de cálculo.
 - Ampliar la disponibilidad de datos.
 - Incremento de fiabilidad.

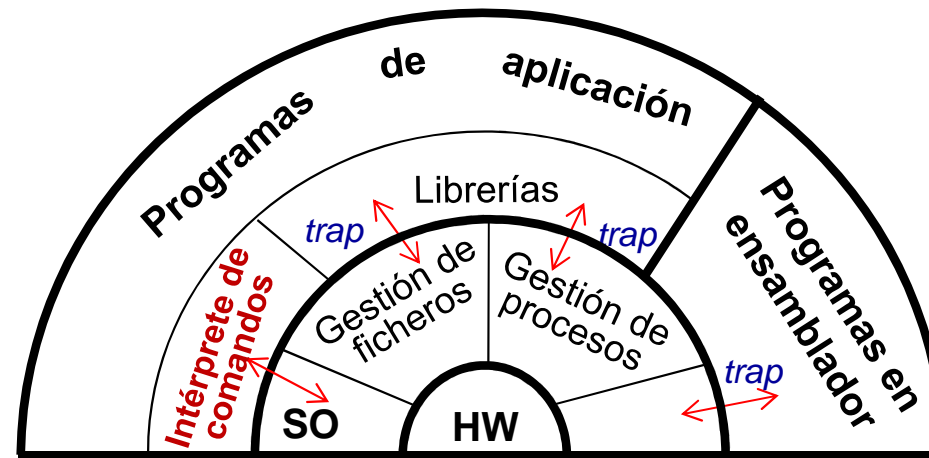


Tema 1. Introducción

Índice

- Concepto de Sistema Operativo (SO)
- Modo supervisor vs. Modo usuario
- Llamadas al sistema
- Interrupciones
- Arranque del SO
- Activación del SO
- Estructura del SO
-  ■ El intérprete de comandos
- Taxonomía de los SO

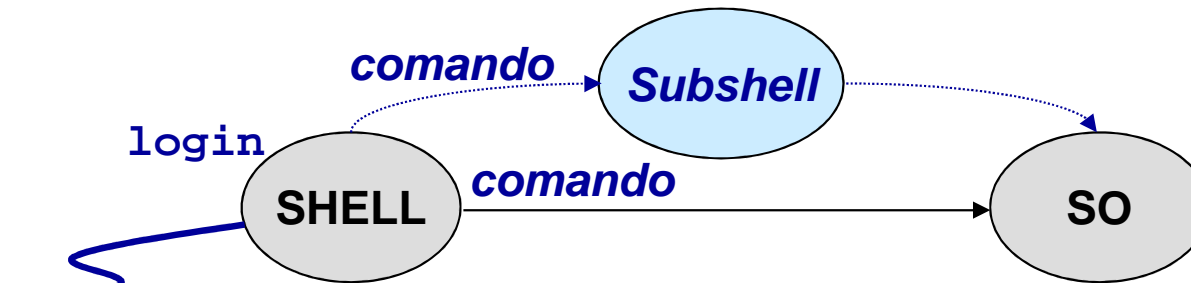
El intérprete de comandos



- **Sistema de intérprete de comandos (intérprete de línea de comandos o shell):**
 - ◆ Concepto de intérprete de comandos:
 - Programa del sistema que lee e interpreta sentencias de control
 - ◆ Función del SO respecto al intérprete de comandos:
 - Obtener y ejecutar cada línea de comandos

El intérprete de comandos

■ Sistema de intérprete de comandos (cont.):



MIENTRAS verdadero HACER

Leer cadena de caracteres desde el terminal

SI no hay errores de sintaxis

ENTONCES

Generar proceso que ejecute la orden

SI (No debe ejecutarse en segundo plano)

Esperar hasta que finalice el proceso generado

FinSI

FinSI

FinMIENTRAS

Llamada al sistema



Tema 1. Introducción

Índice

- Concepto de Sistema Operativo (SO)
- Modo supervisor vs. Modo usuario
- Llamadas al sistema
- Interrupciones
- Arranque del SO
- Activación del SO
- Estructura del SO
- El intérprete de comandos
- Taxonomía de los SO





Taxonomía de los SOs

- **Clasificación en base a la estructura del SO:**
 - ◆ SO monolíticos
 - ◆ SO estructurados
 - SO multinivel (por capas o jerárquico)
 - SO con micronúcleo (modelo cliente/servidor)



Taxonomía de los SOs

■ SOs monolíticos:

- ◆ El SO no tiene una estructura clara y bien definida
- ◆ **El SO es un gran programa compuesto por un conjunto de procedimientos que se ejecutan en un único espacio de direcciones**
Cualquier procedimiento puede llamar a cualquier otro → Desde cualquier parte del SO se puede acceder a cualquier otra directamente
- ◆ El SO trabaja en *modo supervisor* (acceso no restringido)
Los programas de aplicación se ejecutan en *modo usuario* (acceso restringido)
- ◆ Surgen de SO sencillos y pequeños a los que se ha ido añadiendo funcionalidades



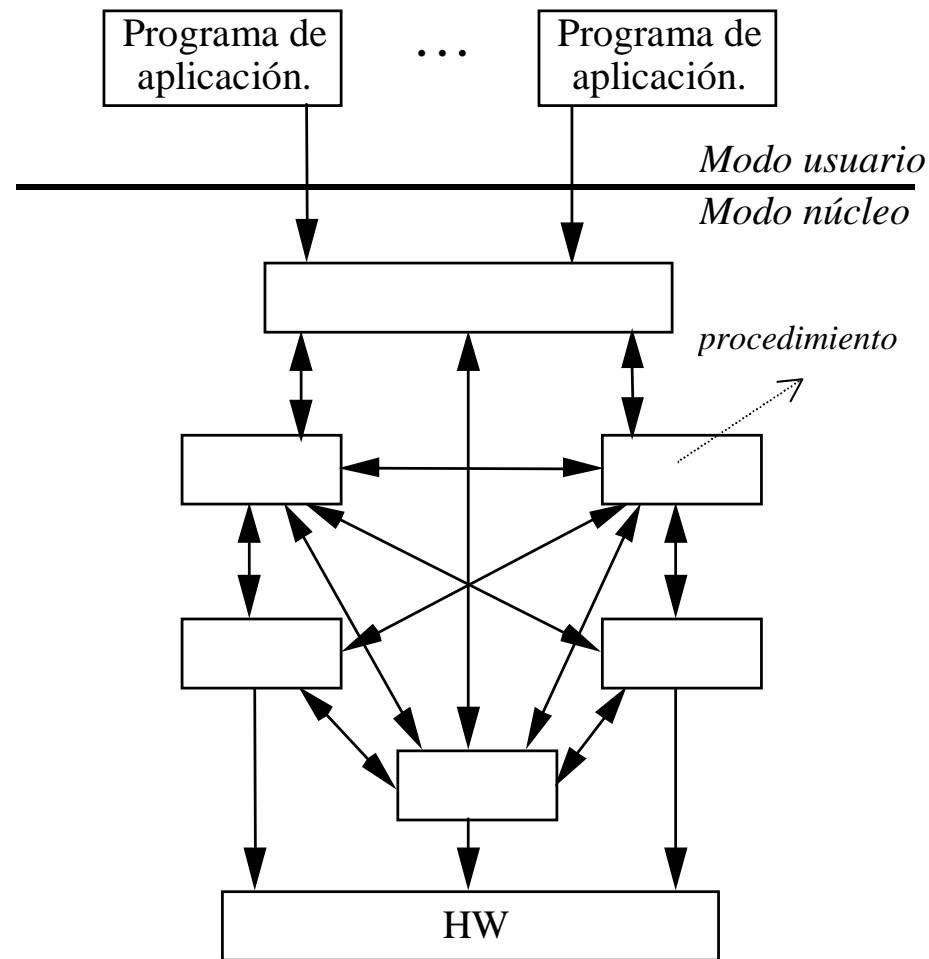
Taxonomía de los SOs

■ SOs monolíticos(cont.):

- ◆ Ventaja:
 - Eficiencia (se producen pocos cambios de contexto)
- ◆ Desventajas:
 - Acceso a todas las estructuras de datos del SO
 - El fallo de un proceso en modo núcleo puede bloquear el SO completo
 - Depuración compleja
 - Extensión del SO compleja
- ◆ Ejemplos de SO monolíticos: MS-DOS, LINUX

Taxonomía de los SOs

■ SOs monolíticos(cont.):





Taxonomía de los SOs

- **SOs multinivel (por capas o jerárquicos):**
 - ◆ El SO se divide en módulos (bloques funcionales bien definidos) que están en diferentes niveles, unos por encima de otros.
 - ◆ El código de un módulo únicamente puede invocar a código de módulos de niveles inferiores o bien a código de niveles inmediatamente anteriores o posteriores.
 - ◆ Cada módulo tiene su propio espacio de direcciones y suele ejecutarse por un proceso diferente.
Paso de un módulo a otro → Cambio de proceso
 - ◆ Ejemplos de SO multinivel: OS/2, THE



Taxonomía de los SOs

■ SOs multinivel (cont.):

◆ Ventajas:

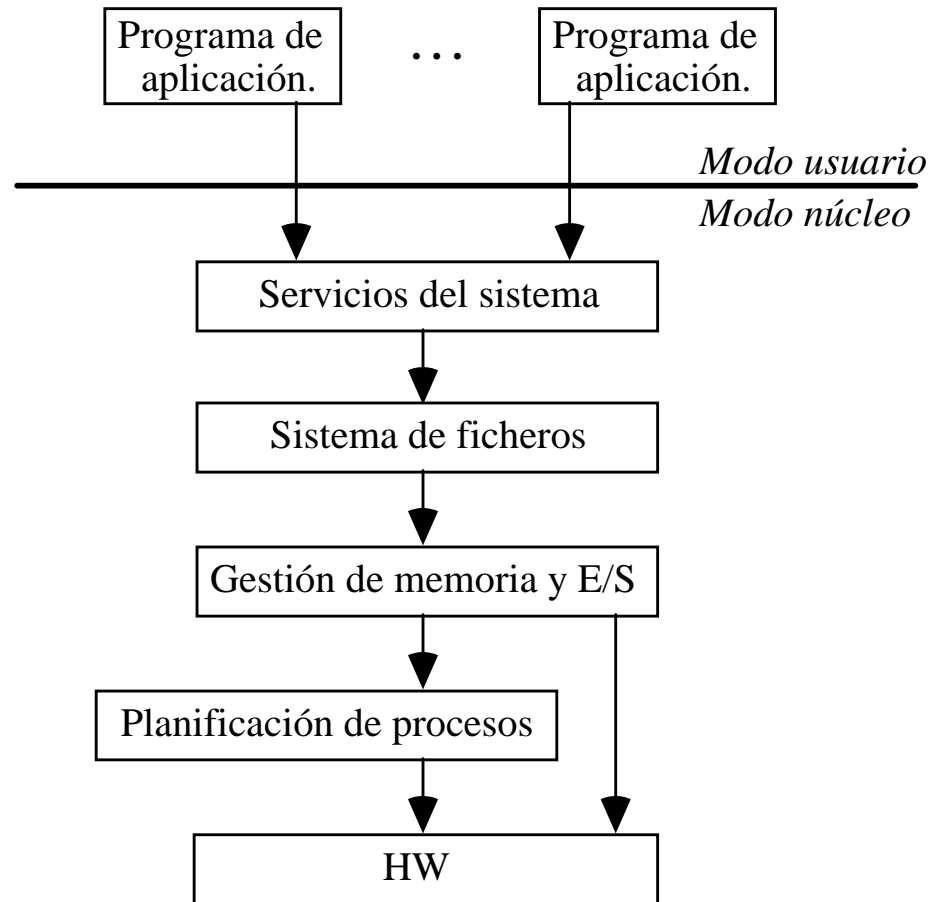
- Acceso limitado a ciertas estructuras de datos del SO.
- Fácil mantenimiento.
- Fácil detección de errores.
- Si falla un módulo (no crítico) no se bloquea todo el SO.
- Extensión del SO sencilla.

◆ Desventaja:

- Eficiencia (se producen muchos cambios de contexto)

Taxonomía de los SOs

■ SOs multinivel (cont.):





Taxonomía de los SOs

- **SOs con micronúcleo (modelo cliente/servidor):**
 - ◆ El SO consta de:
 - Un conjunto de *subsistemas*, cada uno de los cuales implementa un tipo de servicio (servicio de procesos, servicio de memoria, etc.)
 - Un *micronúcleo*, que proporciona la mínima funcionalidad necesaria:
 - ✓ La abstracción de proceso
 - ✓ La gestión de interrupciones
 - ✓ El mecanismo de comunicación entre procesos
 - ◆ Cada subsistema (*servidor*) ejecuta un bucle en el que testea si alguien (*cliente*) ha solicitado su servicio
 - ◆ La petición de servicios y la indicación de la finalización del servicio (junto con el retorno de los resultados) se realiza mediante *paso de mensajes* a través del micronúcleo

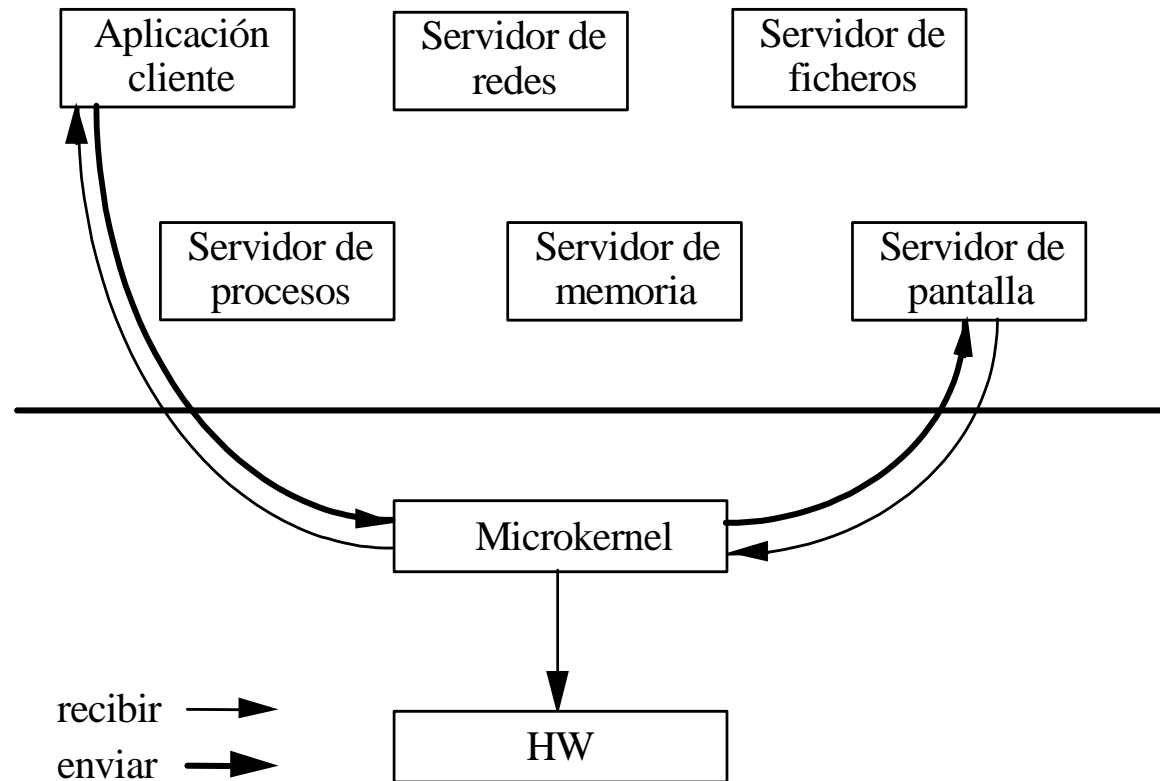


Taxonomía de los SOs

- **SOs con micronúcleo (cont.):**
 - ◆ Desventaja:
 - Poca eficiencia (debido a la sobrecarga de comunicaciones)
 - ◆ Ventajas:
 - Fácil implementación, depuración y mantenimiento
 - Los servidores funcionan independientemente
 - Si falla un servidor (no crítico) no se bloquea todo el SO
 - ✓ Desarrollo sencillo de nuevos servidores
 - ✓ Se puede portar a otras arquitecturas modificando muy poco código
 - ◆ Tendencia actual de los SO
 - ◆ Ejemplos de SO con micronúcleo: SO de Macintosh
 - ◆ Ejemplos de SO multinivel y con micronúcleo: Windows NT, MINIX

Taxonomía de los SOs

■ SOs con micronúcleo (cont.):





Taxonomía de los SOs

- **Tendencias actuales de los SO:**
 - ◆ Modelo cliente/servidor
 - ◆ Arquitecturas micronúcleo
 - ◆ Hilos
 - ◆ Multiprocesamiento (simétrico)
 - ◆ Tecnologías orientadas a objetos
 - ◆ SO distribuidos