

## Capítulo 2

# Introducción a los sistemas operativos

A decorative vertical bar on the left side of the slide, featuring a light red background with faint, overlapping technical diagrams. These diagrams include a network topology at the top, a flowchart in the middle, and a database cylinder at the bottom, all rendered in a darker red color.

# Logros principales

- Los procesos.
- La gestión de memoria.
- La seguridad y la protección de la información.
- La planificación y la gestión de recursos.
- La estructura del sistema.

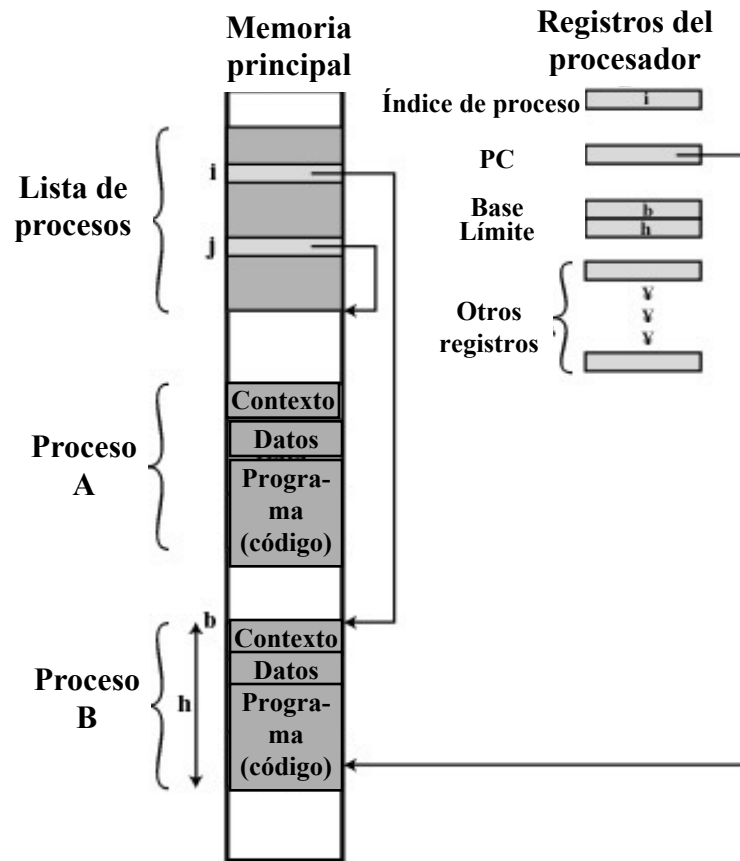
# Procesos

- Un programa en ejecución.
- Una instancia de un programa funcionando en un computador.
- La entidad que puede ser asignada al procesador y ejecutada por él.
- Una unidad de actividad caracterizada por un sencillo tratamiento de ejecución secuencial, un estado actual, y asociada a un conjunto de recursos del sistema.

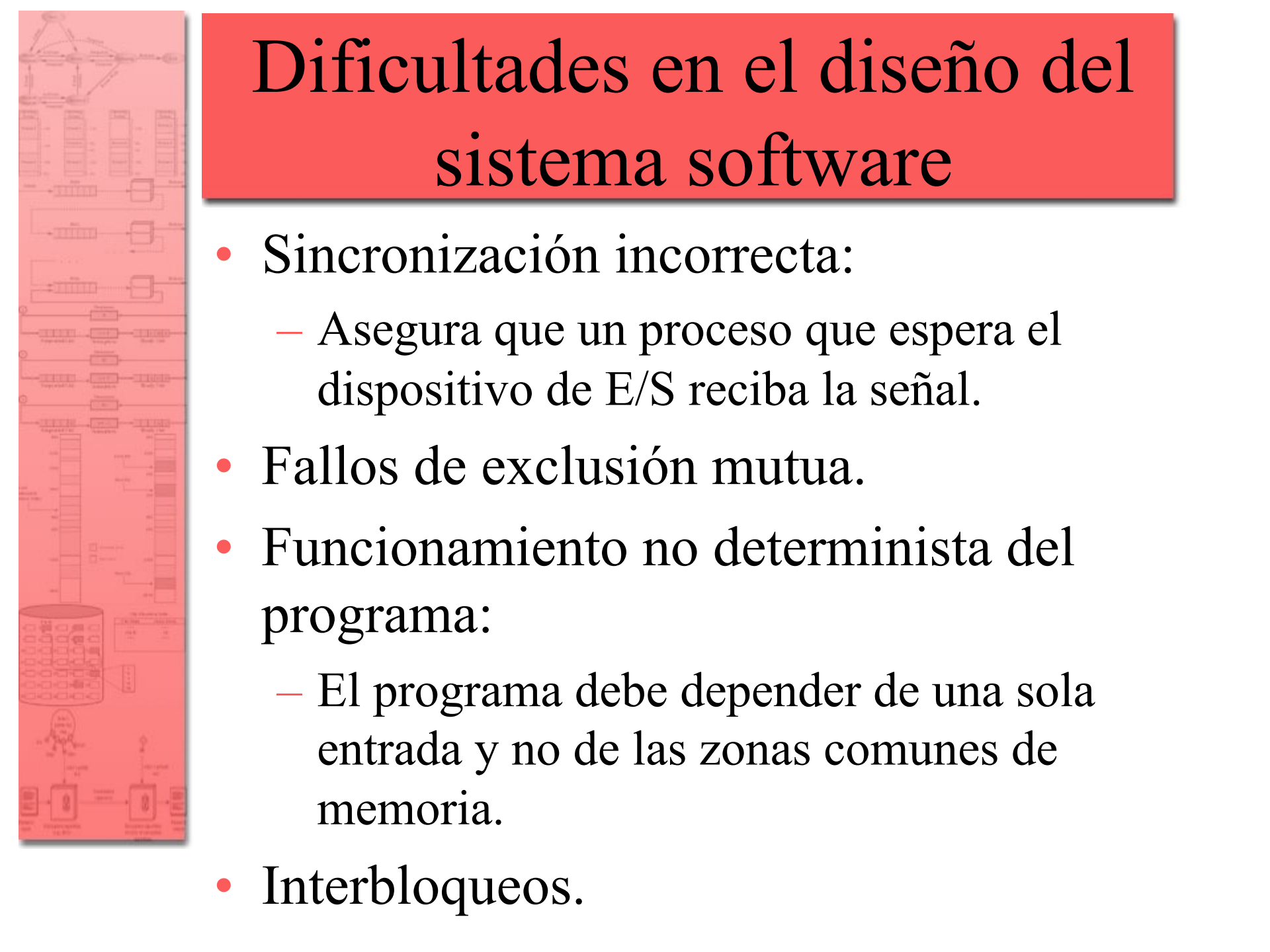
# Proceso

- Formado por tres componentes:
  - Un programa ejecutable.
  - Los datos asociados necesarios para el programa.
  - El contexto de ejecución del programa.
    - Toda la información que el sistema operativo necesita para administrar el proceso.

# Proceso



**Figura 2.8.** Implementación típica de los procesos.



# Dificultades en el diseño del sistema software

- Sincronización incorrecta:
  - Asegura que un proceso que espera el dispositivo de E/S reciba la señal.
- Fallos de exclusión mutua.
- Funcionamiento no determinista del programa:
  - El programa debe depender de una sola entrada y no de las zonas comunes de memoria.
- Interbloqueos.



# Gestión de memoria

- Aislamiento del proceso.
- Asignación y gestión automáticas.
- Soporte para la programación modular.
- Protección y control de acceso.
- Almacenamiento a largo plazo.

# Memoria virtual

- Permite a los programas direccionar la memoria desde un punto de vista lógico.
- No existirá un espacio muerto entre la ejecución de los procesos sucesivos, mientras un proceso se envía al almacenamiento secundario y el proceso que le sucede es traído de éste.





# Sistema de archivos

- Incorpora un almacenamiento a largo plazo.
- La información se almacena en unos objetos denominados archivos.

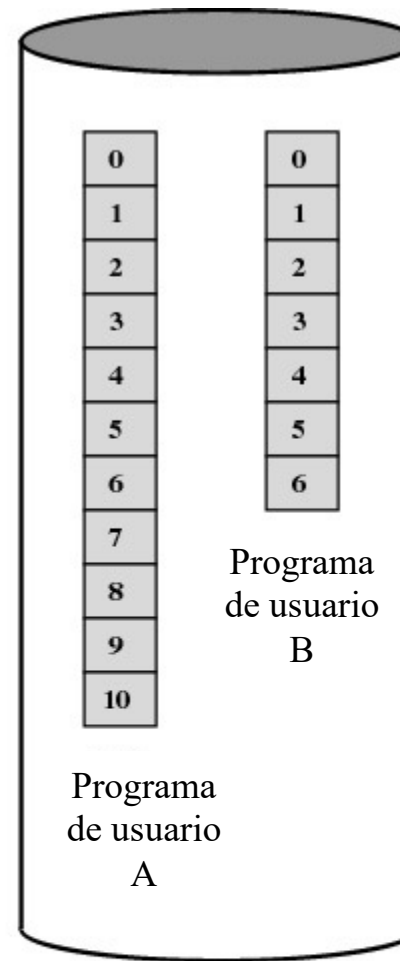
# Sistema de paginación

- Permite que los procesos estén formados por varios bloques de tamaño fijo, denominados páginas.
- La dirección virtual está formada por un número de página y un desplazamiento dentro de la página.
- Cada página puede estar ubicada en un lugar cualquiera de la memoria principal.
- La dirección real o dirección física se utiliza en la memoria principal.

A.1			
	A.0	A.2	
	A.5		
B.0	B.1	B.2	B.3
		A.7	
	A.9		
		A.8	
B.4	B.5	B.6	

**Memoria principal**

La memoria principal está formada por un número de marcos de longitud fija del tamaño de una página. Para ejecutar un programa, algunas o todas sus páginas tienen que estar en el operativo de la memoria principal.

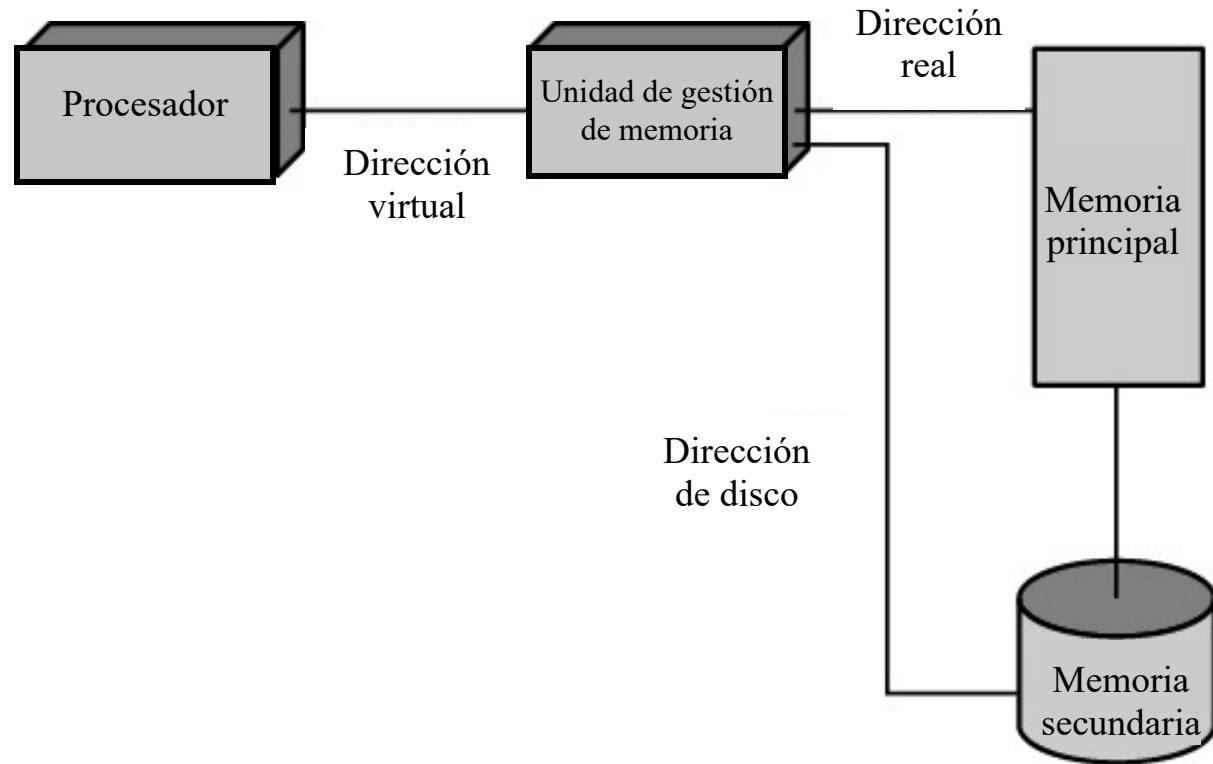


**Disco**

La memoria secundaria (disco) puede almacenar muchas páginas de longitud fija. Un programa de usuario está formado por varias páginas. Las páginas de todos los programas, incluso los del sistema están en el disco, en forma de archivos.

**Figura 2.9.** Conceptos de memoria virtual.

# Direccionamiento de la memoria virtual




**Figura 2.10.** Direccionamiento de la memoria virtual.



# Seguridad y protección de la información

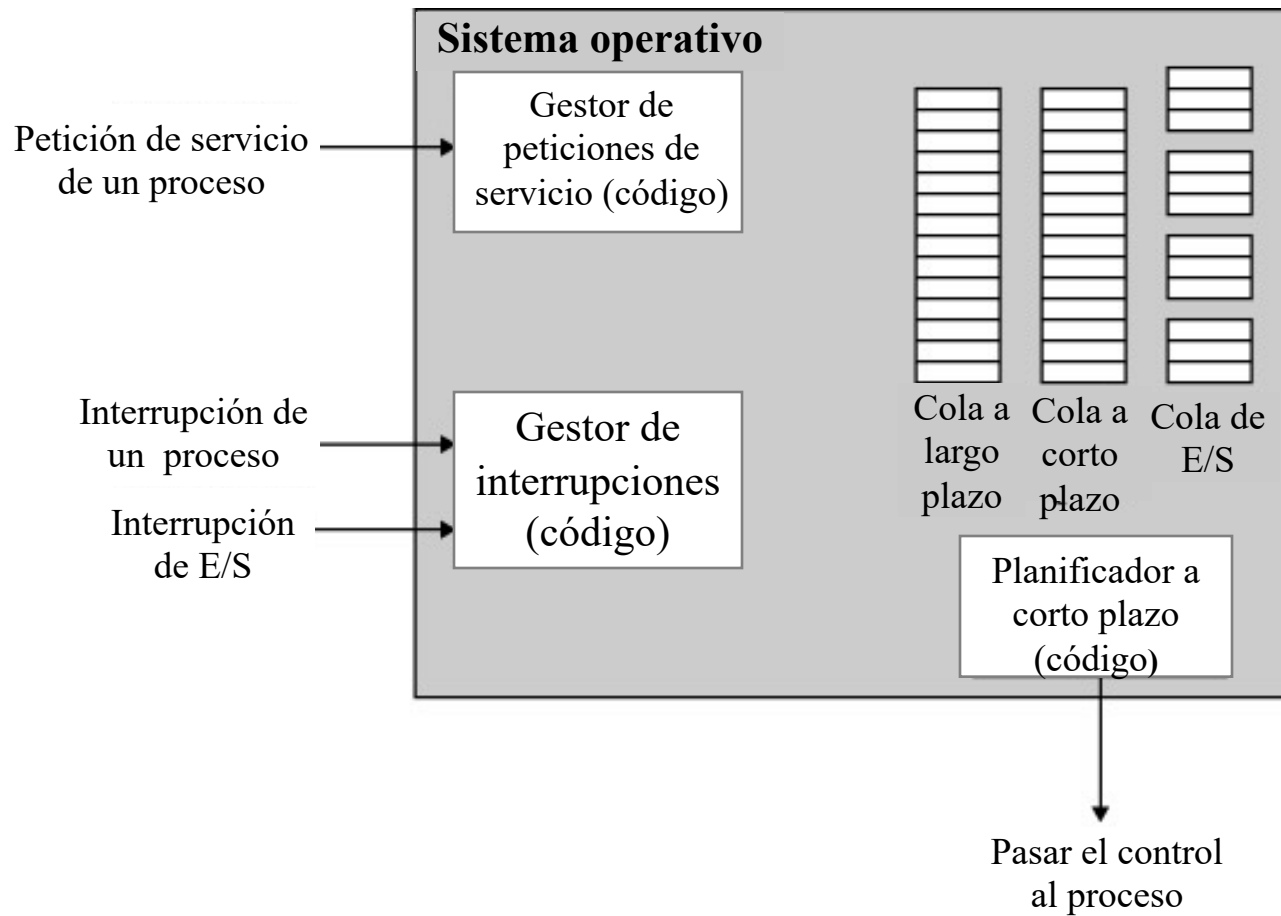
- Control de acceso:
  - Regula el acceso del usuario al sistema.
- Control del flujo de información:
  - Regula el flujo de datos dentro del sistema y su distribución a los usuarios.
- Certificación:
  - Demuestra que el acceso y el control del flujo se llevan a cabo de acuerdo con las especificaciones.



# Planificación y gestión de recursos

- Equidad:
  - Otorga un acceso igual y equitativo a todos los procesos.
- Sensibilidades diferenciales:
  - Discrimina entre las diferentes clases de trabajos.
- Eficiencia:
  - Maximiza la productividad, minimiza el tiempo de respuesta y aloja a tantos usuarios como sea posible.

# Elementos principales de un sistema operativo



**Figura 2.11.** Elementos clave para la multiprogramación de un sistema operativo.



# Estructura del sistema

- Se puede contemplar el sistema como una serie de niveles.
- Cada nivel lleva a cabo un determinado subconjunto de funciones.
- Cada nivel se basa en el nivel inferior para llevar a cabo funciones más primitivas.
- De este modo, se descompone un problema en un número de subproblemas más manejables.



# Jerarquía de diseño de un sistema operativo

Nivel	Nombre	Objetos	Ejemplos de operaciones
13	<i>Shell</i>	Entorno de programación de usuario	Sentencias de un lenguaje de shell
12	Procesos de usuario	Procesos de usuario	Salir, eliminar, suspender, reanudar
11	Directorios	Directorios	Crear, destruir, conectar, desconectar, buscar, listar
10	Dispositivos	Dispositivos externos tales como impresoras, pantallas y teclados	Abrir, cerrar, leer, escribir
9	Sistema de archivos	Archivos	Crear, destruir, abrir, cerrar, leer, escribir
8	Comunicaciones	Tubos ( <i>pipes</i> )	Crear, destruir, abrir, cerrar, leer, escribir

# Jerarquía de diseño de un sistema operativo


Nivel	Nombre	Objetos	Ejemplos de operaciones
7	Memoria virtual	Segmentos, páginas	Leer, escribir, traer ( <i>fetch</i> )
6	Almacenamiento secundario local	Bloques de datos, canales de dispositivos	Leer, escribir, asignar, liberar
5	Procesos primitivos	Procesos primitivos, semáforos, colas de procesos listos	Suspender, reanudar, esperar, señalar

# Jerarquía de diseño de un sistema operativo

Nivel	Nombre	Objetos	Ejemplos de operaciones
4	Interrupciones	Programas de tratamiento de interrupciones	Invocar, enmascarar, desenmascarar, reintentar
3	Procedimientos	Procedimientos, pila de llamadas, visualización	Marcar la pila, llamar, retornar
2	Conjunto de instrucciones	Evaluación de la pila, intérprete de microprogramas, vectores de datos y escalares	Cargar, almacenar, sumar, restar, bifurcar
1	Circuitos electrónicos	Registros, puertas, <i>buses</i> , etc.	Borrar, transferir, activar, complementar


# Características de los sistemas operativos modernos

- Arquitectura micronúcleo:
  - Asigna solamente una pocas funciones esenciales al núcleo.
    - Espacios de direcciones.
    - Comunicación entre procesos (*IPC*).
    - Planificación básica.




# Características de los sistemas operativos modernos

- Multihilos:
  - El proceso se divide en hilos que pueden ejecutarse concurrentemente.
- Hilo:
  - Unidad de trabajo que se puede expedir para su ejecución.
  - Se ejecuta secuencialmente y es interrumpible.
- Proceso:
  - Un conjunto de uno o más hilos.



# Características de los sistemas operativos modernos

- Multiproceso simétrico (*SMP*):
  - Existencia de múltiples procesadores.
  - Estos procesadores comparten la misma memoria principal y dispositivos de E/S.
  - Todos los procesadores pueden ejecutar las mismas funciones.



# Características de los sistemas operativos modernos

- Sistema operativo distribuido:
  - Proporciona la ilusión de un único espacio de memoria principal y un único espacio de memoria secundaria.
  - Utilizado para el sistema de archivos distribuido.

# Características de los sistemas operativos modernos

- Diseño orientado a objetos:
  - Añade extensiones modulares a un pequeño núcleo.
  - Permite a los programadores personalizar un sistema operativo sin romper la integridad del sistema.




# Windows 2000

- Aprovecha las capacidades de los microprocesadores actuales de 32 bits.
- Ofrece una multitarea en un entorno monousuario.
- Proceso cliente/servidor.

# Arquitectura de Windows 2000

- Estructura modular que proporciona flexibilidad.
- Se ejecuta sobre distintas plataformas hardware.
- Soporta aplicaciones escritas para distintos sistemas operativos.



# Organización del sistema operativo (SO)

- Modificación de la arquitectura micronúcleo:
  - No tiene una arquitectura micronúcleo pura.
  - La mayoría de las funciones externas al micronúcleo se ejecutan en modo núcleo.
- Cualquier módulo puede ser eliminado, actualizado o reemplazado sin volver a escribir el sistema completo.

# Estructura por niveles

- Capa de abstracción del hardware (*HAL*):
  - Aísla el sistema operativo de las diferencias de hardware específicas de cada plataforma.
- Micronúcleo:
  - Componentes más usados y fundamentales del sistema operativo.
- Controladores de dispositivo:
  - Traducen las llamadas a funciones de E/S del usuario a peticiones en los dispositivos hardware de E/S específicos.

# Ejecutor de W2K

- Administrador de E/S.
- Administrador de objetos.
- Monitor de seguridad.
- Administrador de procesos/hilos.
- Servicios de llamadas a procedimiento local (*LPC*).
- Administrador de memoria virtual.
- Administrador de cache.
- Módulos para gráficos/ventanas.

# Procesos de usuario

- Procesos de soporte de sistemas especiales.
  - Ejemplo: proceso de inicio de sesión y la gestión de sesión.
- Procesos del servidor.
- Subsistemas de entorno.
- Aplicaciones de usuario.



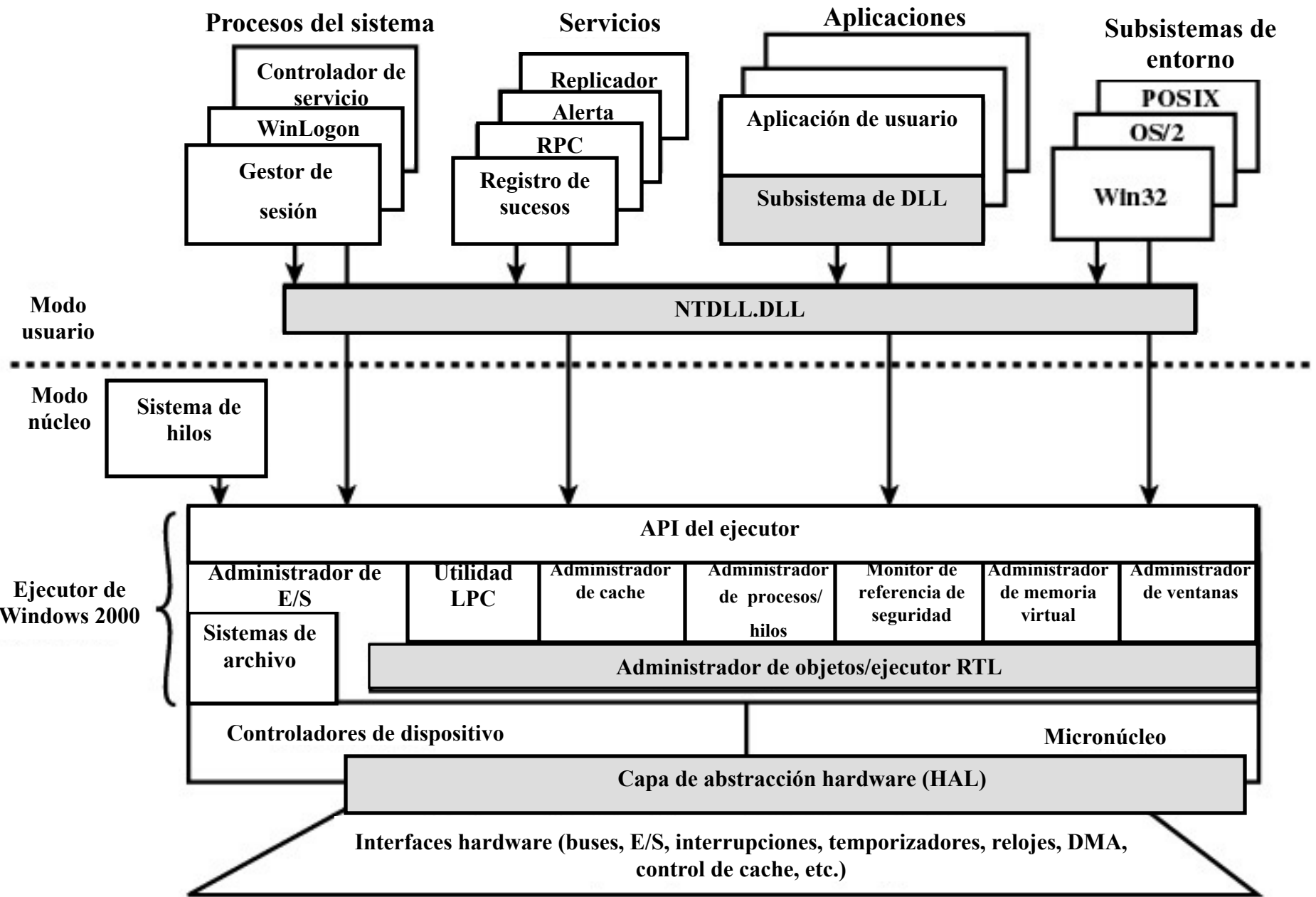
# Modelo cliente/servidor

- Simplifica el Ejecutor:
  - Es posible construir varias API.
- Mejora la fiabilidad:
  - Cada servicio se ejecuta en un proceso independiente, con su propia partición de memoria.
  - Los clientes no pueden acceder directamente al hardware.
- Proporciona un medio uniforme para comunicar las aplicaciones vía LPC.
- Proporciona una base para el procesamiento distribuido.

# Hilos y SMP

- Las diferentes rutinas pueden ejecutar simultáneamente sobre distintos procesadores.
- Los múltiples hilos de ejecución del mismo proceso pueden ejecutar simultáneamente sobre distintos procesadores.
- Los procesos servidor pueden utilizar múltiples hilos.
- Comparten datos y recursos entre procesos.



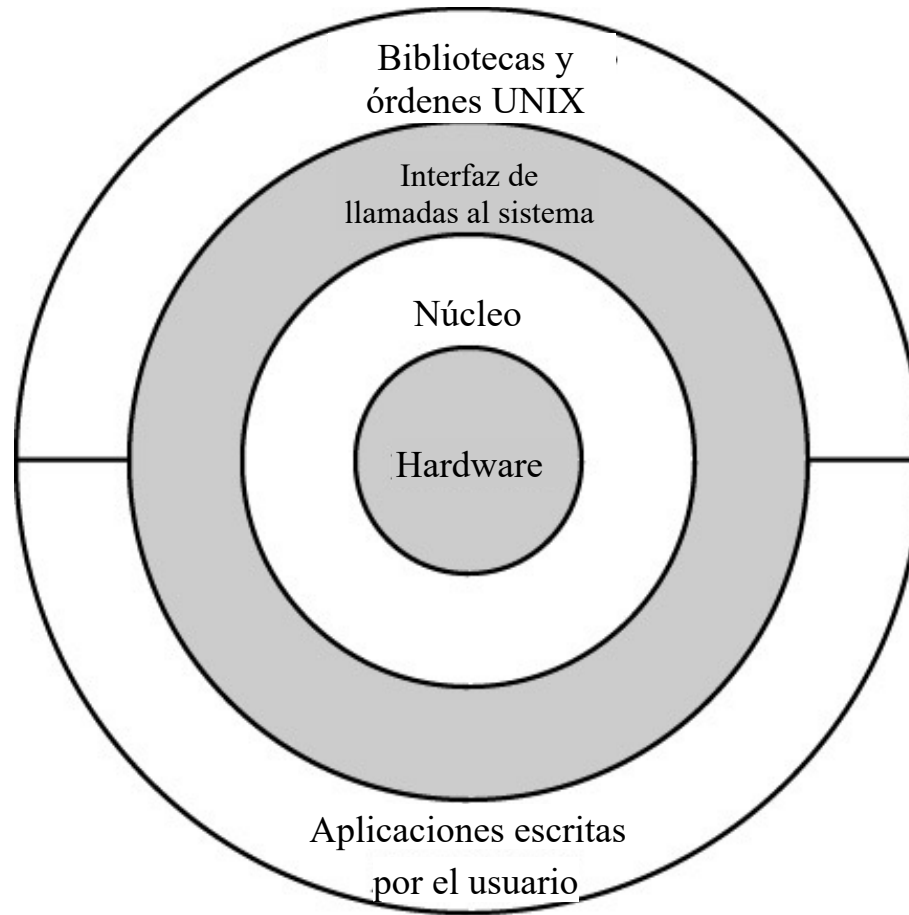


**Figura 2.13.** Arquitectura de Windows 2000.

# UNIX

- El hardware están rodeado por el software del sistema operativo.
- El sistema operativo se llama núcleo (*kernel*).
- Está equipado con varios servicios e interfaces:
  - Intérprete de órdenes (*shell*).
  - Compilador de C.

# UNIX



**Figura 2.15.** Arquitectura general de UNIX.



# Sistemas UNIX modernos

- SVR4
- Solaris 2.x
- 4.4BSD
- Linux