

## Capítulo 4

# Hilos, SMP y micronúcleos



# Proceso

- Unidad de propiedad de los recursos: el proceso incluye un espacio de direcciones virtuales para mantener la imagen del proceso.
- Unidad de expedición: sigue un camino de ejecución que puede ser intercalada con la de otros procesos.
- Estas dos características son tratadas de manera independiente por el sistema operativo.

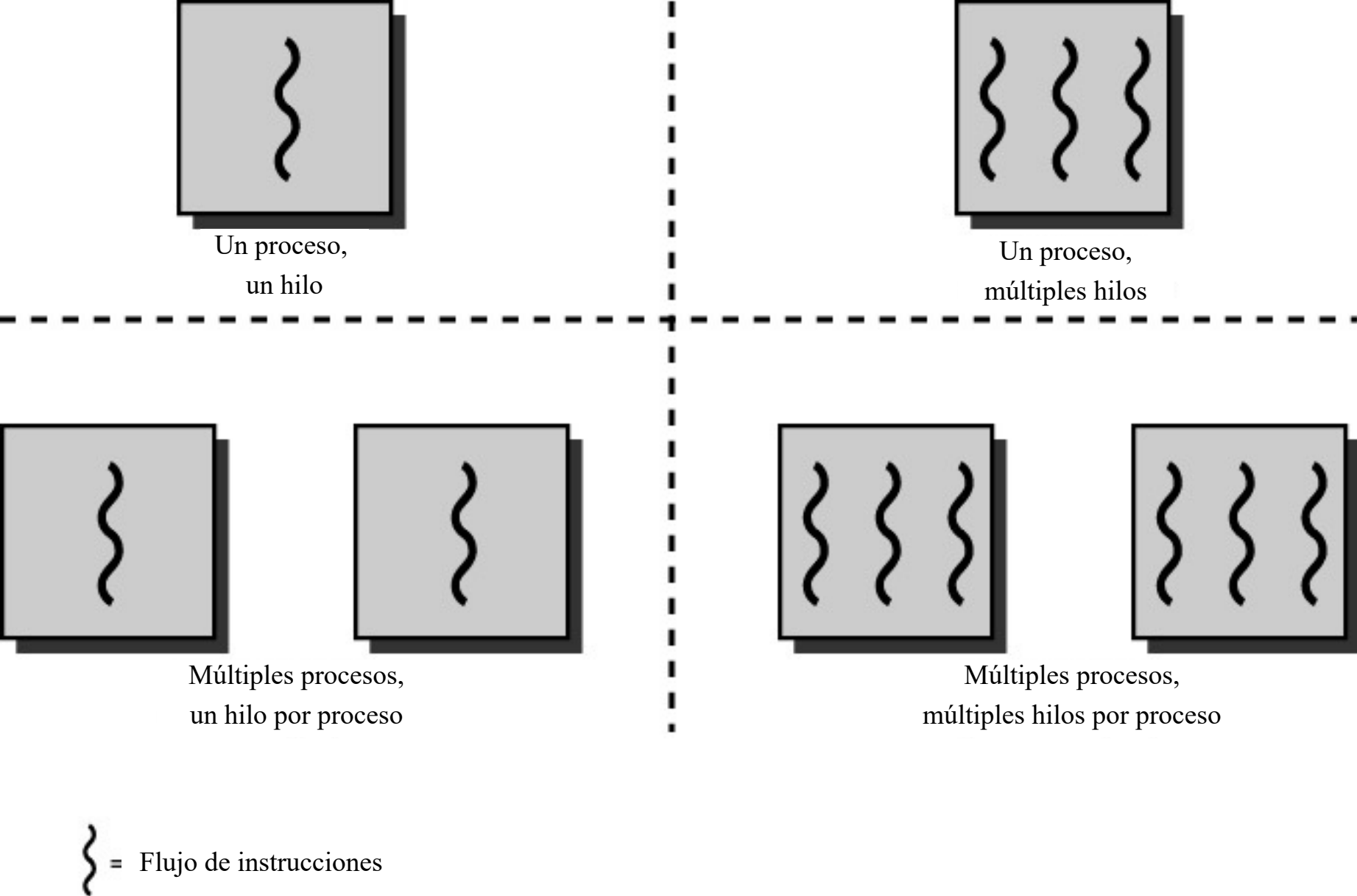


# Proceso

- La unidad de expedición se conoce como hilo.
- La unidad de propiedad de los recursos se conoce como proceso o tarea.

# Multihilo

- Sistema operativo que mantiene varios hilos de ejecución dentro de un mismo proceso.
- MS-DOS soporta un solo hilo.
- UNIX soporta múltiples procesos de usuarios, pero sólo un hilo por proceso.
- Windows 2000, Solaris, Linux, Mach, y OS/2 soportan múltiples hilos.



**Figura 4.1.** Procesos e hilos [ANDE97].

# Proceso

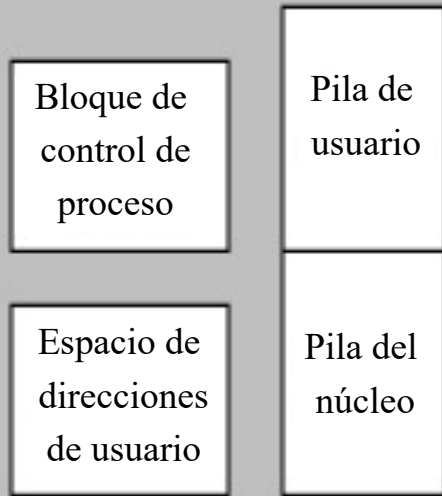
- Tiene un espacio de direcciones virtuales, que contiene la imagen del proceso.
- Acceso protegido a los procesadores, a otros procesos, archivos y a recursos de E/S.



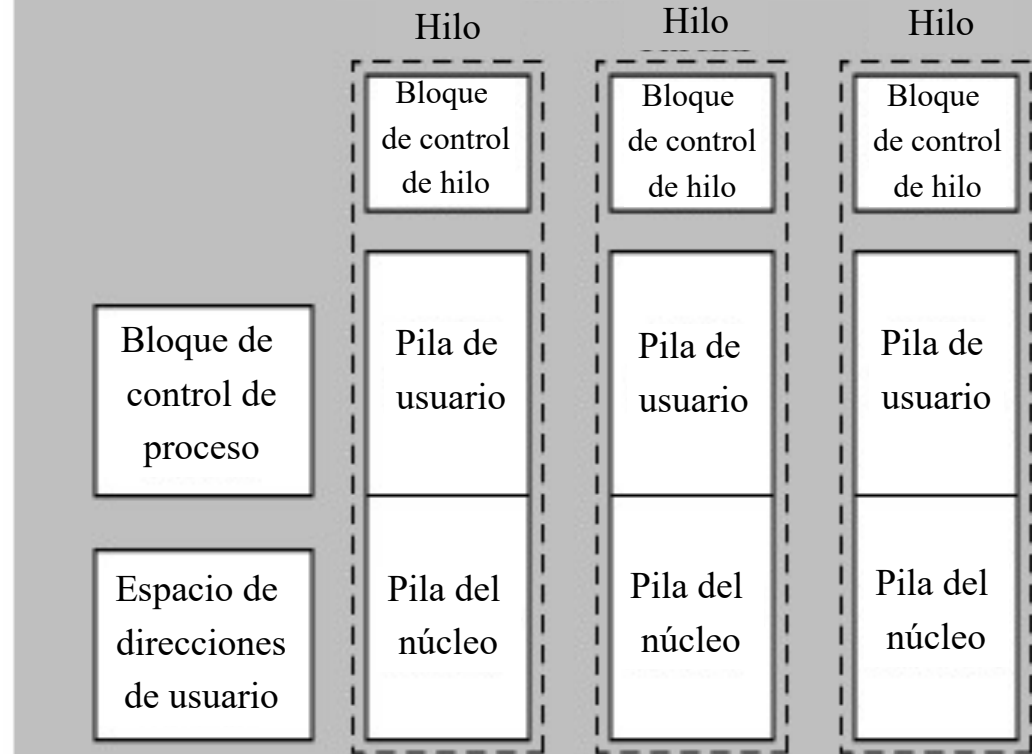
# Hilo

- Posee un estado de ejecución (Ejecución, Listo, etc.).
- El contexto del procesador se salva cuando no está ejecutando.
- Tiene una pila de ejecución.
- Almacenamiento estático para las variables locales.
- Acceso a la memoria y a los recursos del proceso, compartidos con todos los hilos del mismo.

## Modelo de proceso monohilo



## Modelo de proceso multihilo




**Figura 4.2.** Modelos de proceso monohilo y multihilo.





# Beneficios de los hilos

- Se tarda menos tiempo en crear un nuevo hilo en un proceso existente.
- Se tarda menos tiempo en terminar un hilo que un proceso.
- Se tarda menos tiempo en cambiar entre dos hilos de un mismo proceso.
- Puesto que los hilos de un mismo proceso comparten memoria y archivos, pueden comunicarse entre sí sin invocar al núcleo.



# Usos de los hilos en un sistema monousuario y multiproceso

- Trabajo interactivo y en segundo plano.
- Procesamiento asíncrono.
- Aceleración de la ejecución.
- Estructuración modular de los programas.

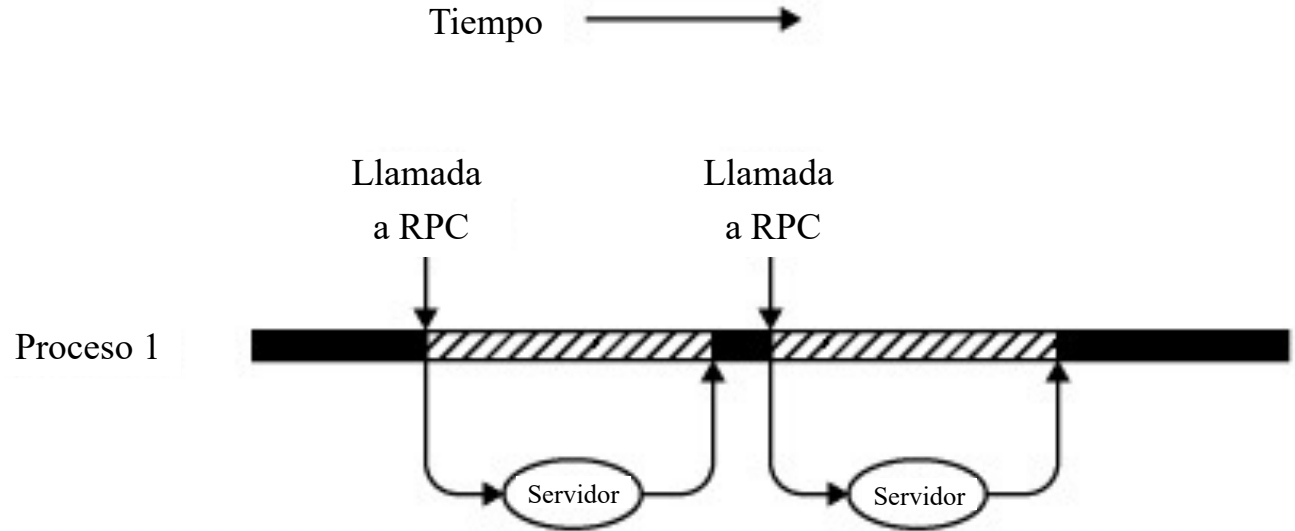
# Hilos

- La suspensión de un proceso implica la suspensión de todos los hilos de un proceso, puesto que todos comparten el mismo espacio de direcciones.
- La terminación de un proceso supone terminar con todos los hilos dentro de dicho proceso.

# Estados de un hilo

- Hay cuatro operaciones básicas relacionadas con el cambio de estado en hilos:
  - Creación:
    - Se crea un nuevo hilo.
  - Bloqueo.
  - Desbloqueo.
  - Terminación:
    - Se liberan el contexto y las pilas.

# Llamadas a Procedimiento Remoto (RPC) utilizando hilos



(a) RPC utilizando un solo hilo



Bloqueado, esperando respuesta de una RPC



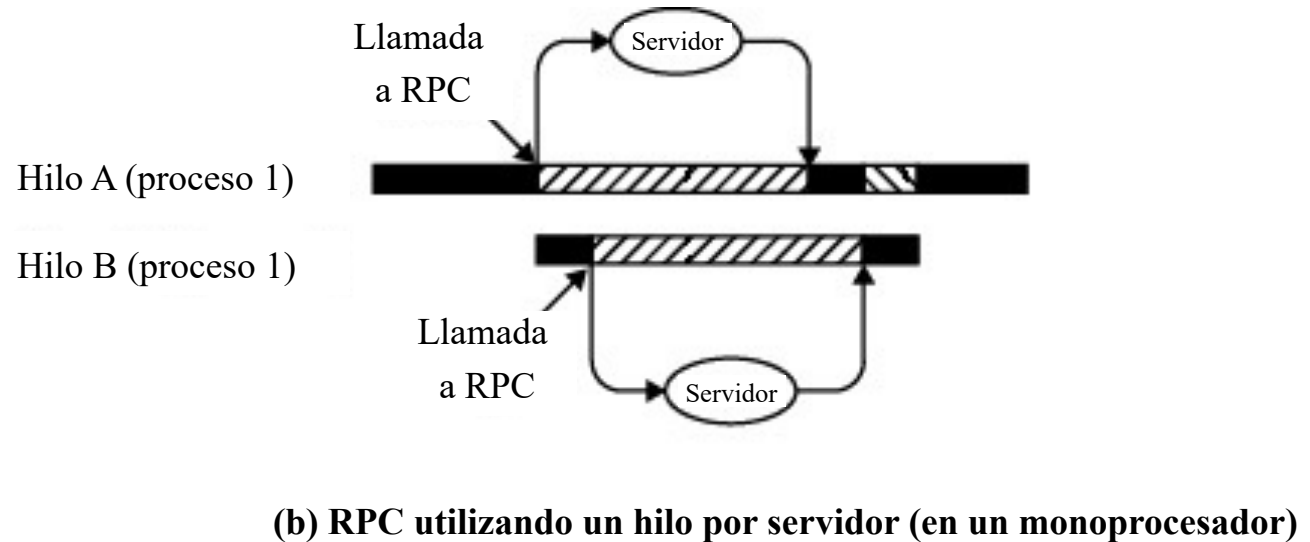
Bloqueado, esperando al procesador que está siendo usado por el hilo B






Ejecutando

**Figura 4.3.** Llamadas a Procedimiento Remoto (RPC) utilizando hilos.

# Llamadas a Procedimiento Remoto (RPC) utilizando hilos



-  Bloqueado, esperando respuesta de una RPC
-  Bloqueado, esperando al procesador que está siendo usado por el hilo B
-  Ejecutando

**Figura 4.3.** Llamadas a Procedimiento Remoto (RPC) utilizando hilos.



# Hilos a nivel de usuario

- La aplicación realiza todo el trabajo de gestión de hilos.
- El núcleo no tiene conocimiento de la existencia de hilos.

# Hilos a nivel de núcleo

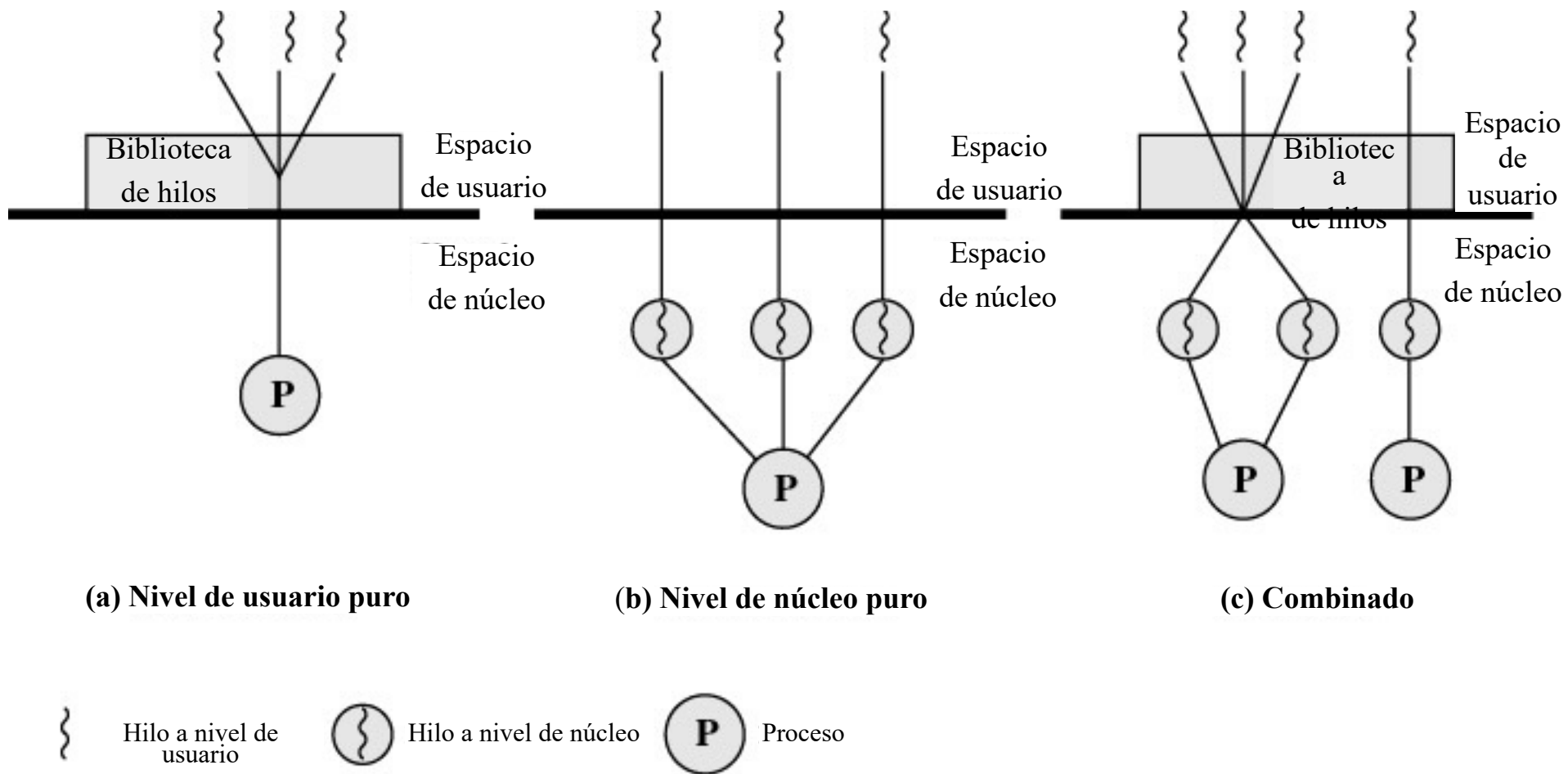
- W2K, Linux y OS/2 son ejemplos de este tipo de aplicación.
- El núcleo mantiene la información de contexto del proceso y de los hilos.
- La planificación se realiza en función de los hilos.





# Aproximaciones combinadas

- Un ejemplo es Solaris.
- La creación de hilos se realiza en el espacio de usuario.
- La planificación y sincronización de los hilos se realiza en el espacio de usuario.



**Figura 4.6.** Hilos a nivel de usuario y a nivel de núcleo.



# Relación entre hilos y procesos

Hilos : Procesos	Descripción	Sistemas de ejemplo
<b>1 : 1</b>	Cada hilo de ejecución es un único proceso con sus propios recursos y espacio de direcciones.	Implementaciones UNIX clásicas
<b>M : 1</b>	Un proceso define un espacio de direcciones y unos recursos dinámicos propios. Pueden crearse varios hilos que ejecuten en dicho proceso.	Windows NT, Solaris, OS/2, OS/390, MACH



# Relación entre hilos y procesos

Hilos : Procesos	Descripción	Sistemas de ejemplo
<b>1 : M</b>	Un hilo puede emigrar del entorno de un proceso a otro. Esto permite que un hilo se pueda mover fácilmente entre sistemas distintos.	Ra (Clouds), Emerald
<b>M : N</b>	Combina los atributos de los casos M : 1 y 1 : M	TRIX

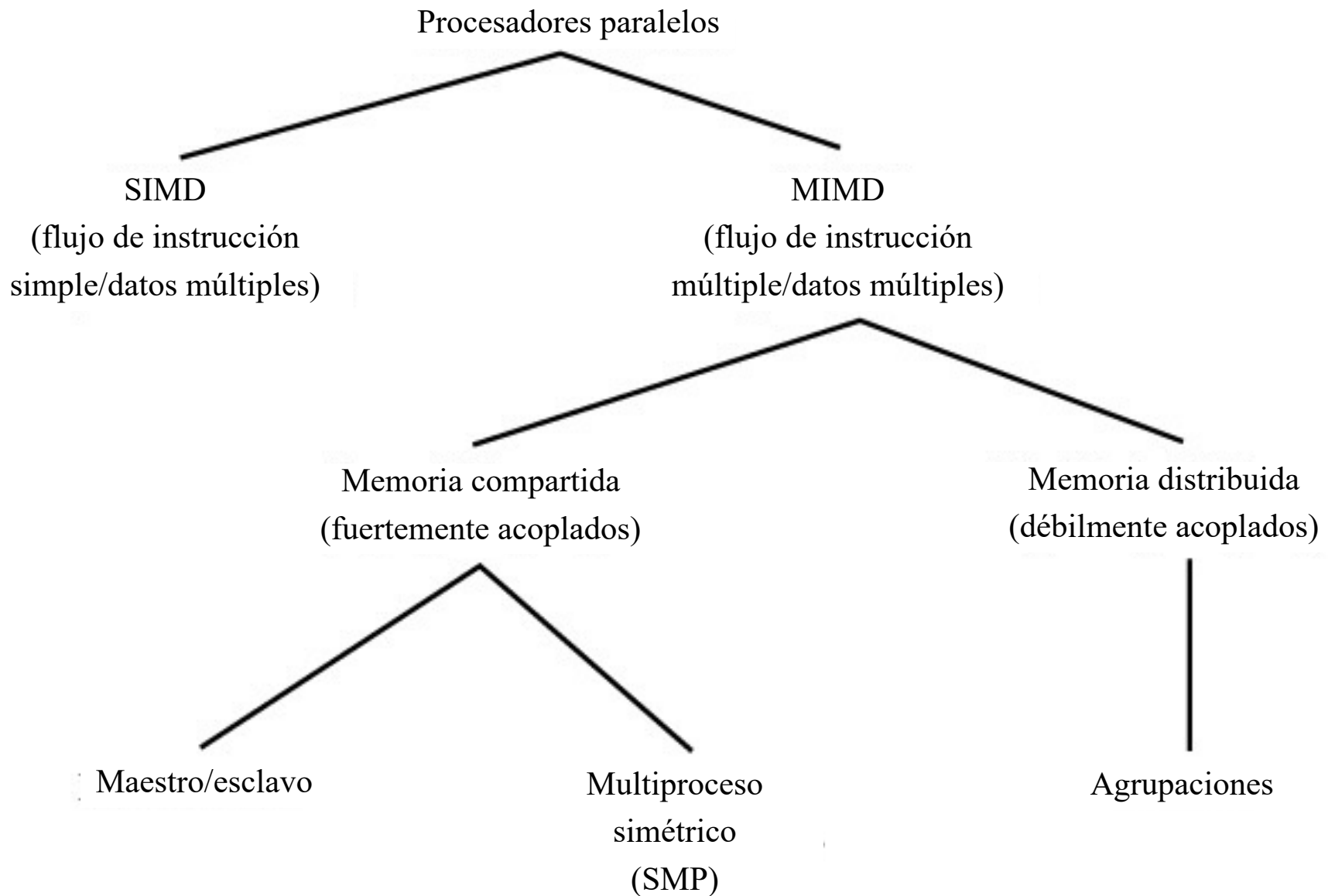


# Categorías de sistemas informáticos

- Instrucción simple/dato simple (SISD):
  - Un único procesador ejecuta un único flujo de instrucciones para operar sobre datos almacenados en una única memoria.
- Instrucción simple/datos múltiples (SIMD):
  - Cada instrucción se ejecuta sobre un conjunto de datos diferente por medio de distintos procesadores.

# Categorías de sistemas informáticos

- Instrucción múltiple/dato simple (MISD):
  - Se transmite una secuencia de datos a un conjunto de procesadores, cada uno de los cuales ejecuta una instrucción de la secuencia. No se ha implementado nunca.
- Instrucción múltiple/datos múltiples (MIMD):
  - Un conjunto de procesadores ejecuta simultáneamente varias secuencias de instrucciones sobre distintos conjuntos de datos.

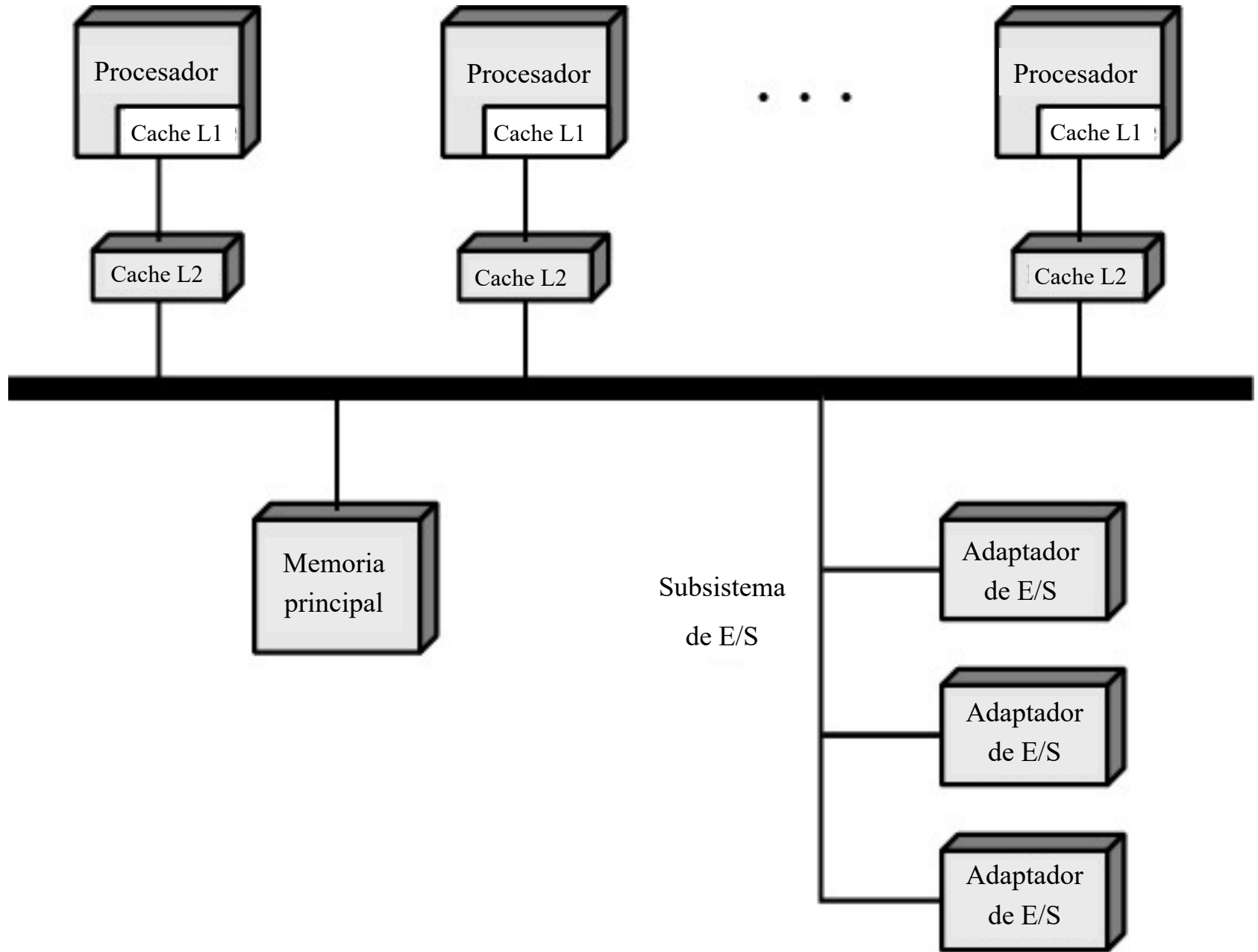


**Figura 4.8.** Arquitecturas de procesadores paralelos.


# Multiproceso simétrico

- El núcleo puede ejecutar en cualquier procesador.
- Normalmente, cada procesador se autoplanifica a partir de una cola de procesos o hilos libres.





**Figura 4.9.** Organización de un multiprocesador simétrico.



# Consideraciones de diseño de un sistema operativo multiprocesador

- Procesos o hilos concurrentes.
- Planificación.
- Sincronización.
- Gestión de memoria.
- Fiabilidad y tolerancia a los fallos.

# Micronúcleos

- Pequeño núcleo del sistema operativo.
- Sólo contiene las funciones esenciales del sistema operativo.
- Muchos de los servicios que tradicionalmente se incluían en el sistema operativo son ahora subsistemas externos:
  - Controladores de dispositivos.
  - Sistemas de archivos.
  - Gestores de memoria virtual.
  - Sistemas de ventanas.
  - Servicios de seguridad.



# Ventajas de la organización micronúcleo

- Interfaz uniforme para las solicitudes realizadas por los procesos:
  - Todos los servicios se utilizan mediante paso de mensajes.
- Extensibilidad:
  - Permite añadir nuevos servicios.
- Flexibilidad:
  - Permite añadir nuevas características.
  - Permite reducir las características actuales.

# Ventajas de la organización micronúcleo

- Portabilidad:
  - En el micronúcleo (y no en los demás servicios) se realizan los cambios necesarios para portar el sistema a un nuevo procesador.
- Fiabilidad:
  - Diseño modular.
  - Un pequeño micronúcleo puede probarse de un modo muy riguroso.

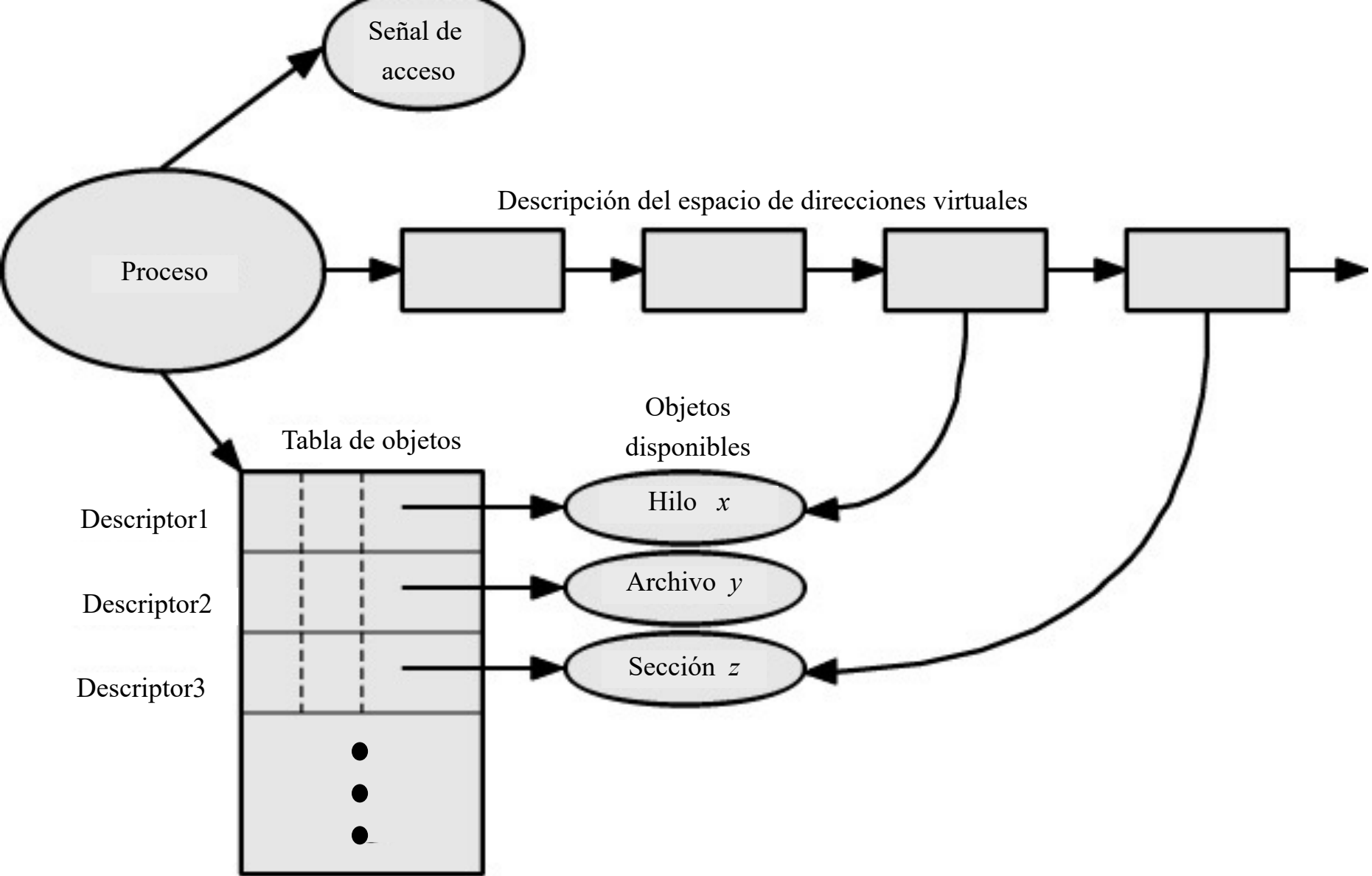


# Ventajas de la organización micronúcleo

- Soporte a sistemas distribuidos:
  - Se puede enviar un mensaje sin saber en qué máquina reside el destinatario.
- Sistema operativo orientado a objetos:
  - Los componentes son objetos con interfaces claramente definidas que se pueden interconectar para formar un software.

# Diseño de micronúcleo

- Gestión de memoria a bajo nivel:
  - Traduce cada página virtual en un marco de página físico.
- Comunicación entre procesos.
- Gestión de interrupciones y E/S.



**Figura 4.12.** Un proceso de Windows 2000 y sus recursos [CUST93].



# Objeto proceso en Windows 2000

Tipo de objeto

Proceso

Atributos  
del cuerpo  
del objeto

ID del proceso  
Descriptor de seguridad  
Prioridad de base  
Afinidad por omisión con el procesador  
Límites de cuota  
Tiempo de ejecución  
Contadores de E/S  
Contadores de operación de la MV  
Puertos de excepciones y depuración  
Estado de terminación

Servicios

Crear proceso  
Abrir proceso  
Consultar información del proceso  
Cambiar información del proceso  
Proceso actual  
Terminar proceso

(a) Objeto proceso



# Objeto hilo en Windows 2000

Tipo de objeto

Hilo

Atributos  
del cuerpo  
del objeto

ID del hilo  
Contexto del hilo  
Prioridad dinámica  
Prioridad de base  
Afinidad del hilo con el procesador  
Tiempo de ejecución del hilo  
Estado de alerta  
Contador de suspensión  
Señal de imitación  
Puerto de terminación  
Estado de terminación del hilo

Servicios

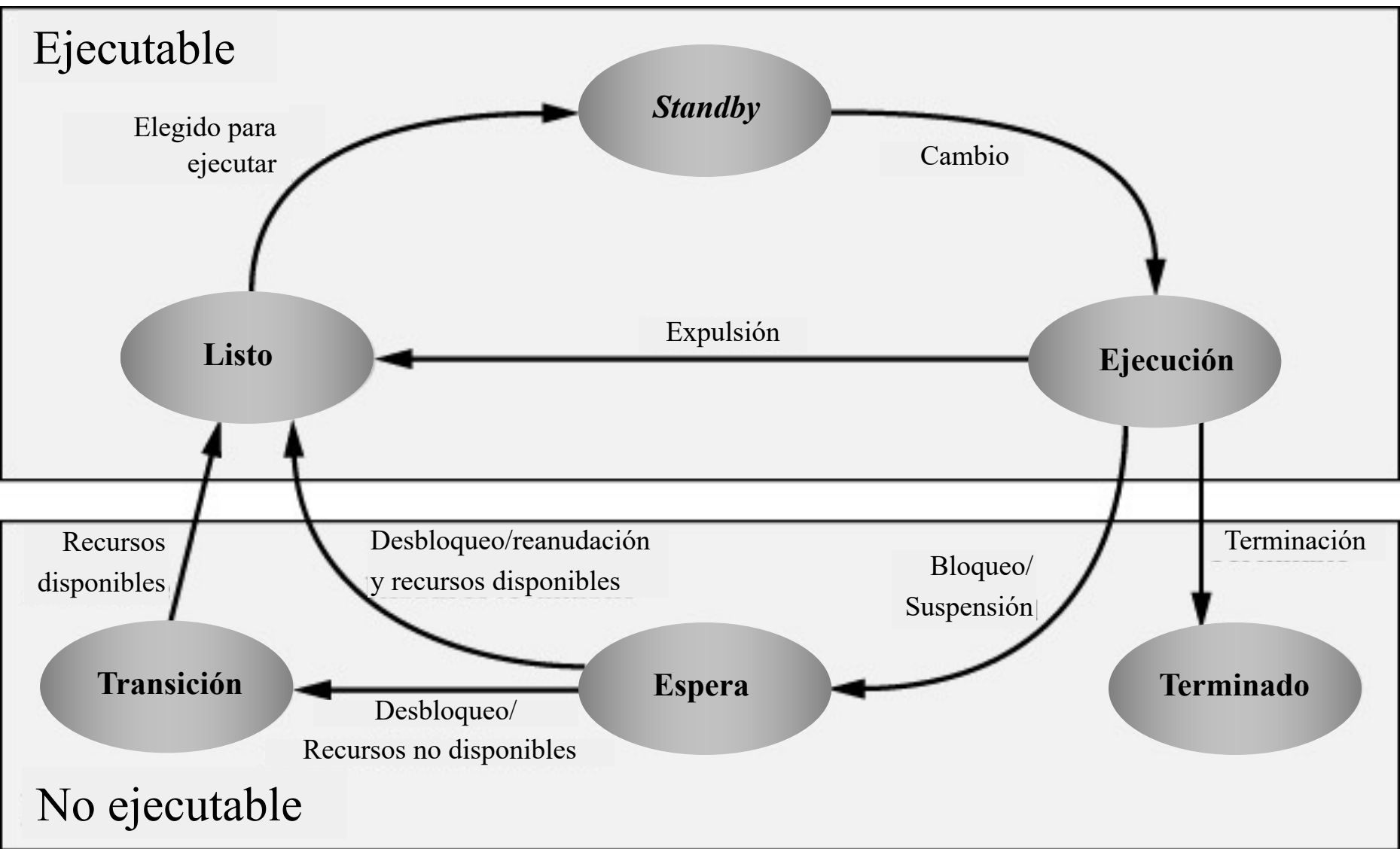
Crear hilo  
Abrir hilo  
Consultar información del hilo  
Cambiar información del hilo  
Hilo actual  
Terminar hilo  
Coger contexto  
Poner contexto  
Suspender  
Reanudar  
Alertar hilo  
Consultar alerta del hilo  
Registrar puerto de terminación

**(b) Objeto hilo**

# Estados de un hilo en Windows 2000

- Listo.
- *Standby*.
- Ejecución.
- Espera.
- Transición.
- Terminado.

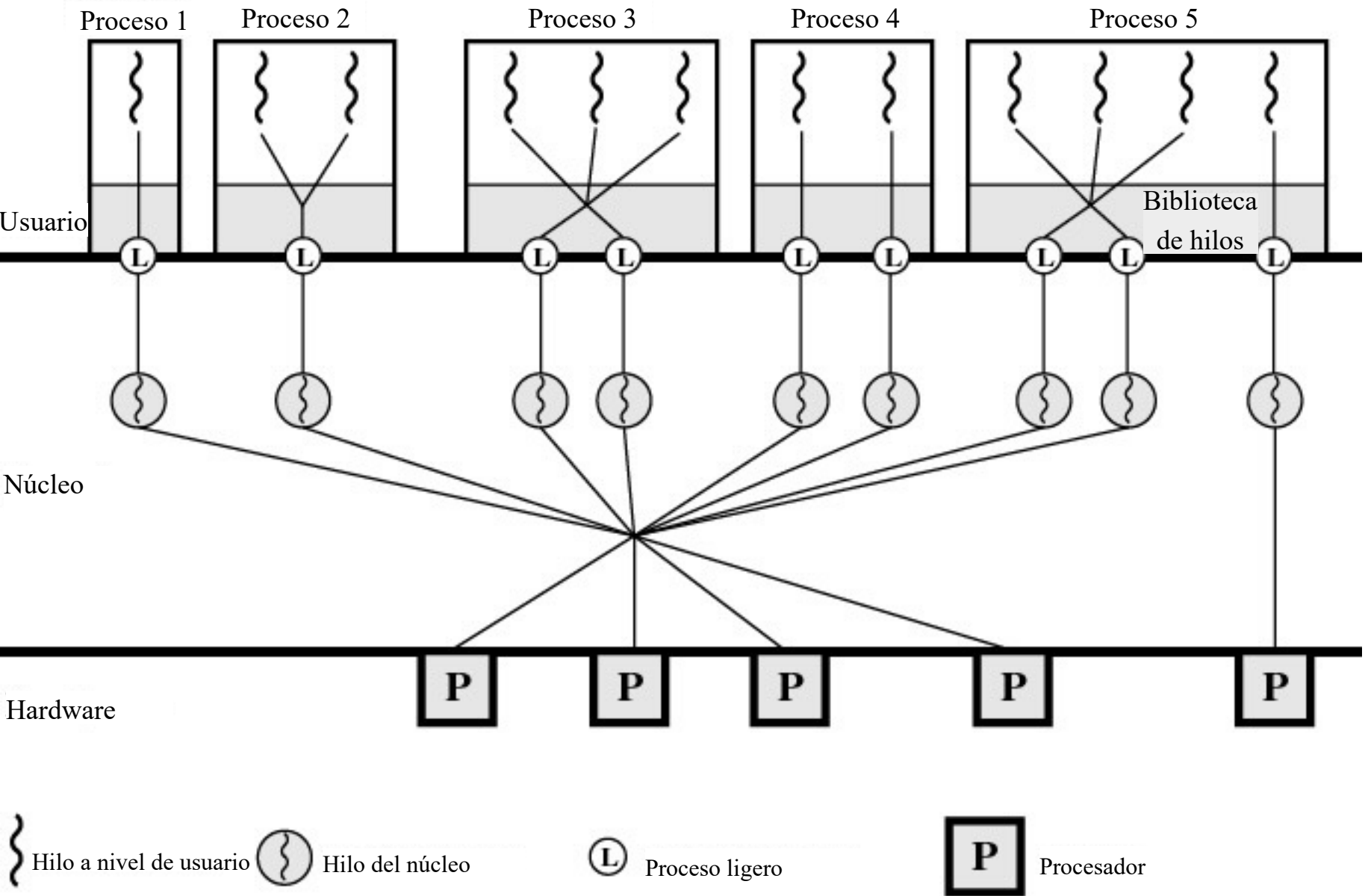




**Figura 4.14.** Estado de hilos en Windows 2000.

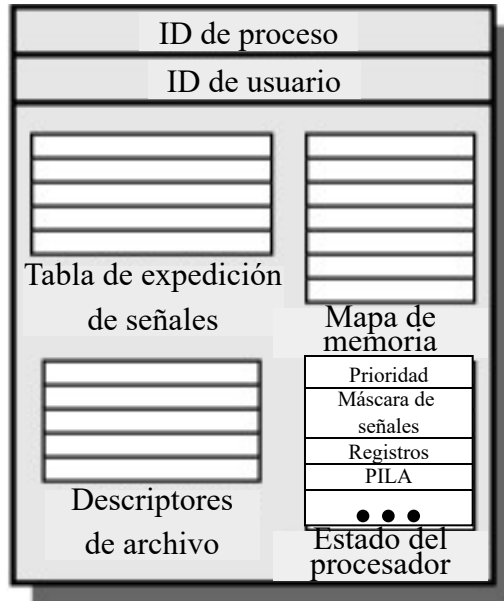
# Solaris

- El proceso incluye el espacio de direcciones de usuario, la pila y el bloque de control de proceso.
- Hilos a nivel de usuario.
- Procesos ligeros.
- Hilos de núcleo.

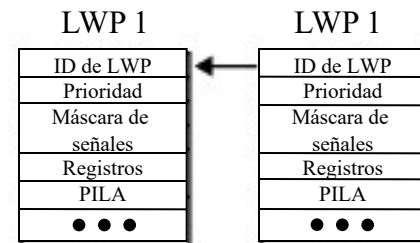
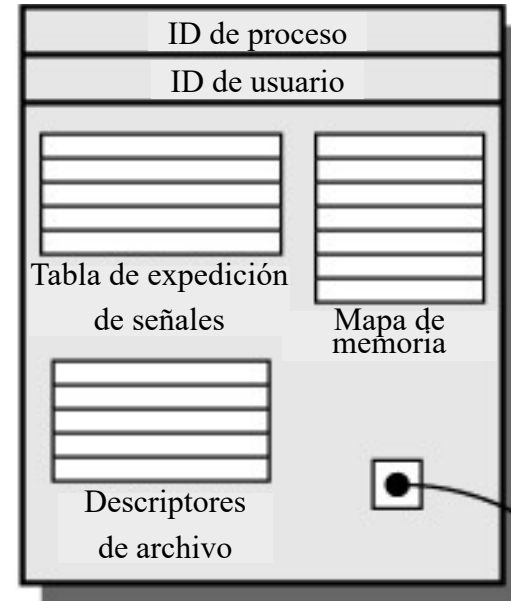


**Figura 4.15.** Ejemplo de la arquitectura multihilo de Solaris.

## Estructura de un proceso en UNIX



## Estructura de un proceso en Solaris 2.x



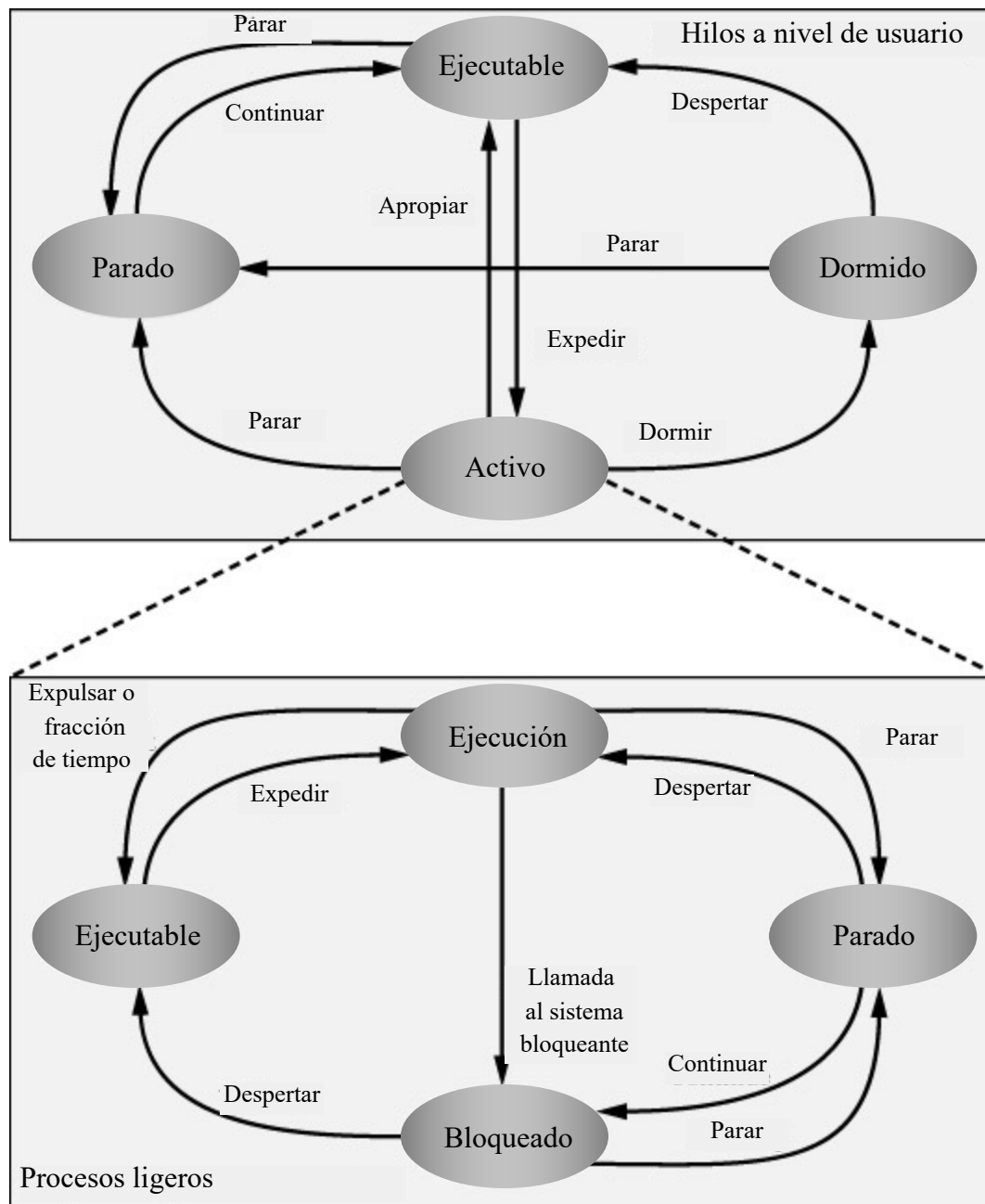
**Figura 4.16.** Estructura de un proceso en UNIX clásico y Solaris 2.x [LEWI96].



# Ejecución de hilos en Solaris

- Sincronización.
- Suspensión.
- Apropiación.
- Cesión.





**Figura 4.17.** Estados de LWP e hilos a nivel de usuario en Solaris.

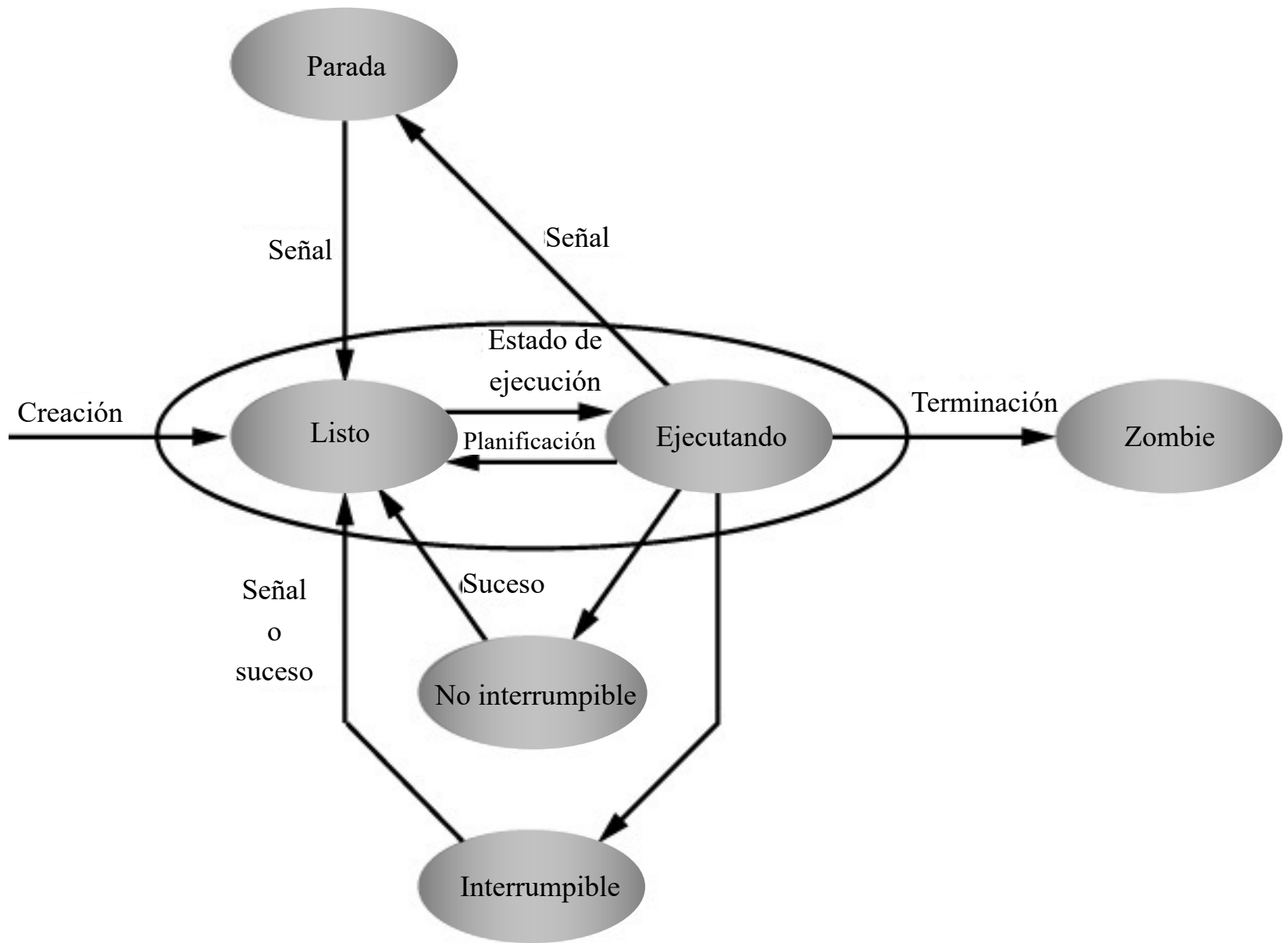


# Procesos en Linux

- Estado.
- Información de planificación.
- Identificadores.
- Comunicación entre procesos.
- Vínculos.
- Tiempos y temporizadores.
- Sistema de archivos.
- Memoria virtual.
- Contexto específico del procesador.

# Estados de un proceso en Linux

- Ejecución.
- Interrumpible.
- No interrumpible.
- Parado.
- Zombie.



**Figura 4.18.** Modelo de procesos/hilos en Linux.