

# SISTEMAS OPERATIVOS

Capítulo 3. Descripción y control de procesos -  
planificación

# Requisitos de un sistema operativo

2

- ❑ Intercala la ejecución de múltiples procesos para maximizar la utilización del procesador mientras se proporciona un tiempo de respuesta razonable
- ❑ Reserva recursos para los procesos
- ❑ Da soporte a la comunicación entre procesos y la creación de procesos

# Conceptos

3

- Una plataforma de computación consiste en una colección de recursos hardware
- Las aplicaciones para computadores se desarrollan para realizar determinadas tareas
- No es eficiente que las aplicaciones estén escritas directamente para una plataforma hardware específica
- El sistema operativo proporciona una interfaz apropiada para las aplicaciones, rica en funcionalidades, segura y consistente
- El sistema operativo proporciona una representación uniforme y abstracta de los recursos que las aplicaciones pueden solicitar y acceder

# Gestión de la ejecución de aplicaciones

4

- Los recursos están disponibles para múltiples aplicaciones
- El procesador se conmuta entre múltiples aplicaciones
- El procesador y los dispositivos de E/S se pueden usar de forma eficiente

# Proceso

5

- Un programa en ejecución
- Una instancia de un programa ejecutado en un computador
- La entidad que se puede asignar y ejecutar en un procesador
- Una unidad de actividad que se caracteriza por la ejecución de una secuencia de instrucciones, un estado actual y un conjunto de recursos del sistema asociados

# Bloque de control de proceso

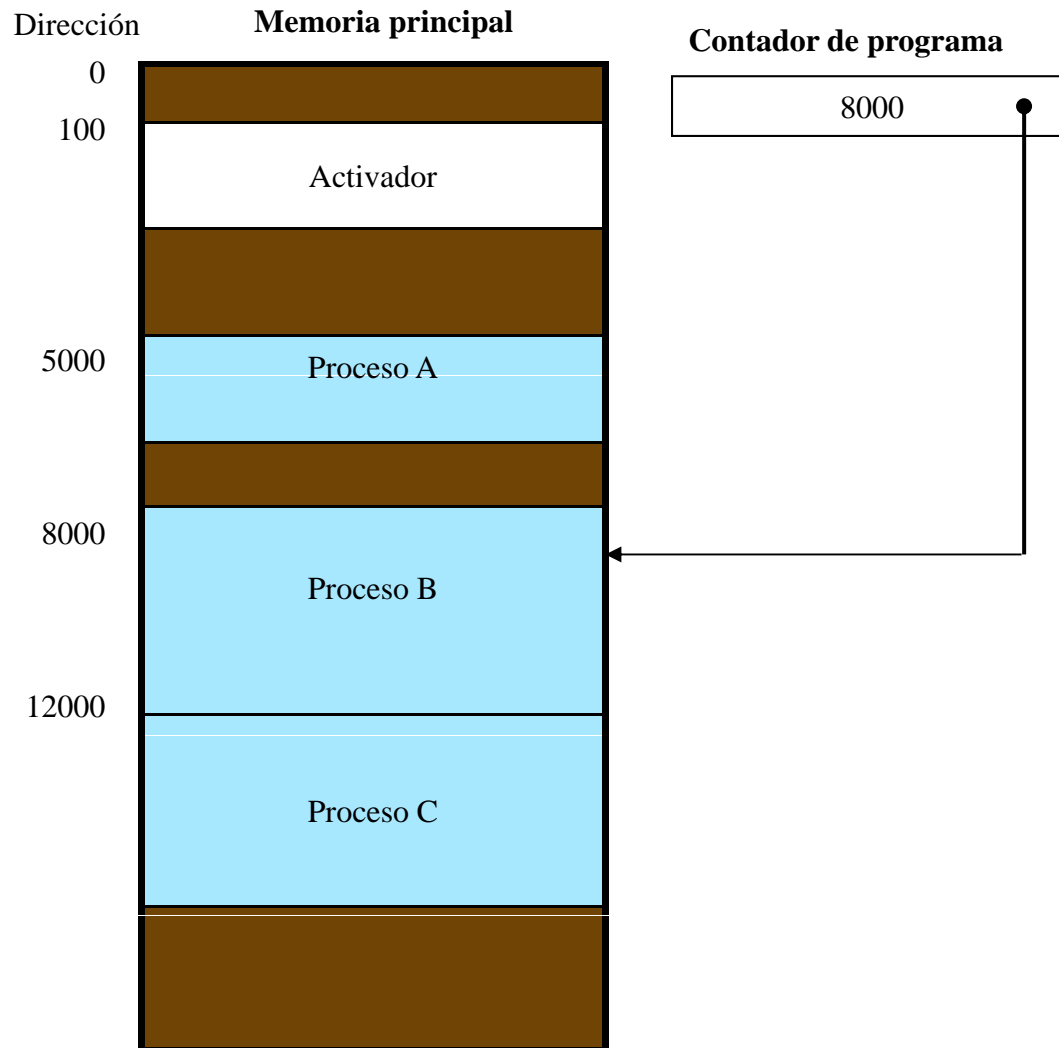
6

- Contiene los elementos del proceso
- Creado y gestionado por el sistema operativo
- Permite dar soporte a múltiples procesos

Identificador
Estado
Prioridad
Contador de programa
Punteros de memoria
Datos de contexto
Información de estado de E/S
Información de auditoría
• • •

# Traza del proceso

7



- Secuencia de instrucciones que se ejecutan para dicho proceso
- Un pequeño programa activador (*dispatcher*) intercambia el procesador de un proceso a otro

# Creación de procesos /razones

8

## **Nuevo proceso de lotes**

El sistema operativo dispone de un flujo de control de lotes de trabajos, habitualmente una cinta o disco. Cuando el sistema operativo está listo para procesar un nuevo trabajo, leerá la siguiente secuencia de mandatos de control de trabajos.

## **Sesión interactiva**

Un usuario desde un terminal entra en el sistema.

## **Creado por el sistema operativo para proporcionar un servicio**

El sistema operativo puede crear un proceso para realizar una función en representación de un programa de usuario, sin que el usuario tenga que esperar (por ejemplo, un proceso para controlar la impresión)

## **Creado por un proceso existente**

Por motivos de modularidad o para explotar el paralelismo, un programa de usuario puede ordenar la creación de un número de procesos.



# Terminación de procesos/razones

9

<b>Finalización normal</b>	El proceso ejecuta una llamada al sistema operativo para indicar que ha completado su ejecución .
<b>Límite de tiempo excedido</b>	El proceso ha ejecutado más tiempo del especificado en un límite máximo. Existen varias posibilidades para medir dicho tiempo. Incluyen el tiempo total utilizado, el tiempo utilizado únicamente en ejecución y en el caso de procesos interactivos, la cantidad de tiempo desde que el usuario realizó la última entrada.
<b>Memoria no disponible</b>	El proceso requiere más memoria de la que el sistema puede proporcionar.
<b>Violaciones de frontera</b>	El proceso trata de acceder a una posición de memoria a la cual no tiene acceso permitido.
<b>Error de protección</b>	El proceso trata de usar un recurso, por ejemplo un fichero, al que no tiene permitido acceder, o trata de utilizarlo de una forma impropia, como escribiendo en un fichero de sólo lectura.
<b>Error aritmético</b>	El proceso trata de realizar una operación de cálculo no permitida, como una división entre 0, o trata de almacenar números mayores de los que la representación hardware puede codificar.

# Terminación de procesos/razones

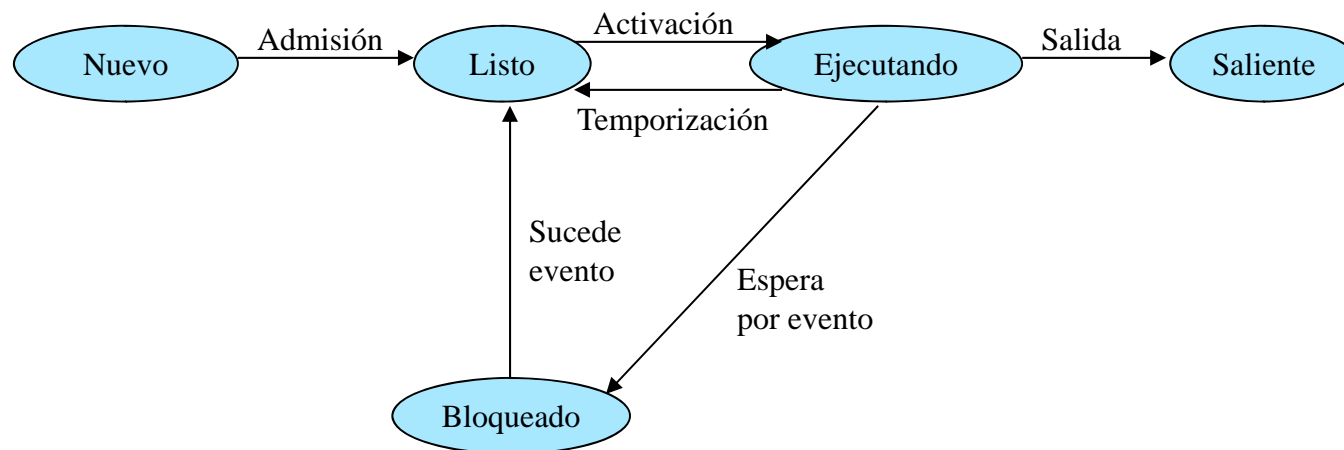
10

<b>Límite de tiempo</b>	El proceso ha esperado más tiempo que el especificado en un valor máximo para que se cumpla un determinado evento.
<b>Fallo de E/S</b>	Se ha producido un error durante una operación de entrada o salida, por ejemplo la imposibilidad de encontrar un fichero, fallo en la lectura o escritura después de un límite máximo de intentos (cuando, por ejemplo, se encuentra un área defectuosa en una cinta), o una operación inválida (la lectura de una impresora en línea).
<b>Instrucción no válida</b>	El proceso intenta ejecutar una instrucción inexistente (habitualmente el resultado de un salto a un área de datos y el intento de ejecutar dichos datos).
<b>Instrucción privilegiada</b>	El proceso intenta utilizar una instrucción reservada al sistema operativo.
<b>Uso inapropiado de datos</b>	Una porción de datos es de tipo erróneo o no se encuentra inicializada.
<b>Intervención del operador por el sistema operativo</b>	Por alguna razón, el operador o el sistema operativo ha finalizado el proceso (por ejemplo, se ha dado una condición de interbloqueo).
<b>Terminación del proceso padre</b>	Cuando un proceso padre termina, el sistema operativo puede automáticamente finalizar todos los procesos hijos descendientes de dicho padre.
<b>Solicitud del proceso padre</b>	Un proceso padre habitualmente tiene autoridad para finalizar sus propios procesos descendientes.

# Procesos / Estados

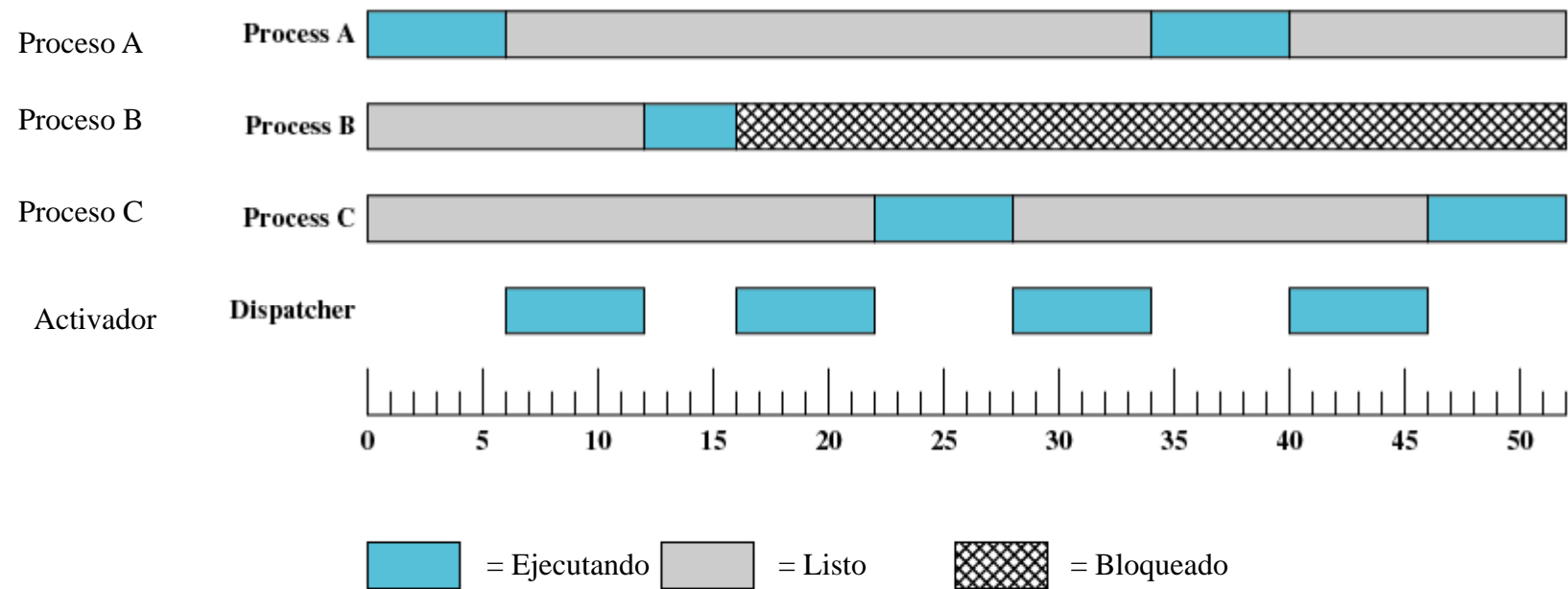
11

- Ejecutando
- Listo
  - ▣ Listo para ejecutar
- Bloqueado
  - ▣ Esperando a que se complete una operación de E/S
- El activador no puede seleccionar únicamente el proceso que lleve más tiempo en la cola ya que puede estar bloqueado



# Estado de los procesos

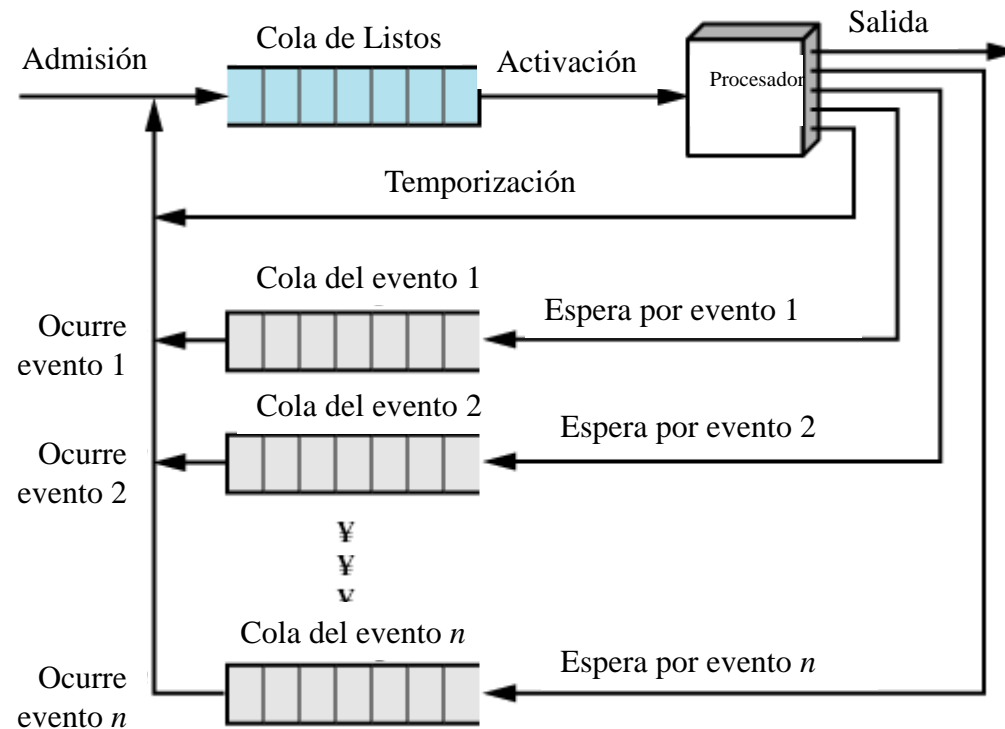
12



Ejemplo Estado de procesos

# Múltiples colas de bloqueados

13



(b) Múltiples colas de Bloqueado

-Modelo de colas de la Figura -----

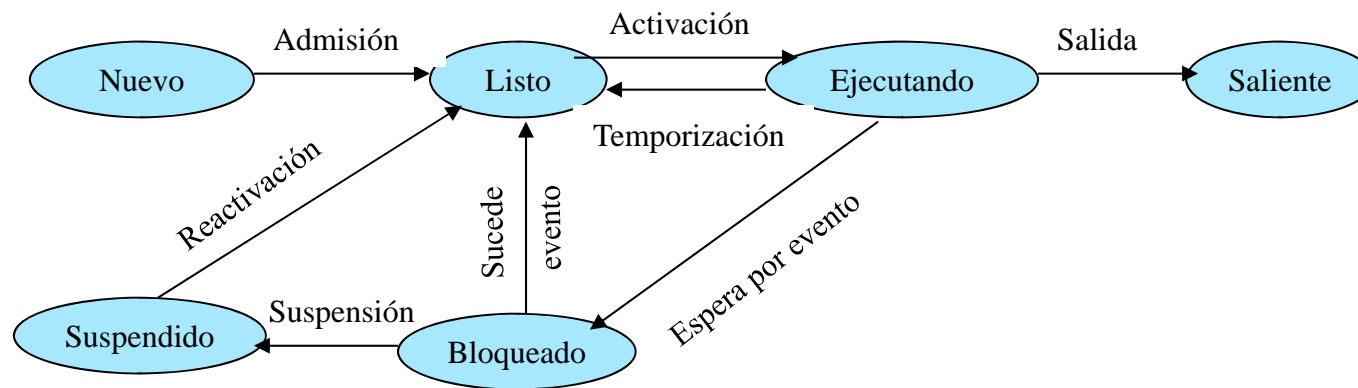
# Procesos suspendidos

14

- El procesador es más rápido que las operaciones de E/S, por lo que todos los procesos estarían a la espera de dichas operaciones
- Mover estos procesos al disco para expandir la memoria
- El estado Bloqueado de un proceso se convierte en estado Suspendido cuando se transfiere al disco
- Dos nuevos estados
  - ▣ Bloqueado/Suspendido
  - ▣ Listo/Suspendido

# Un único estado suspendido

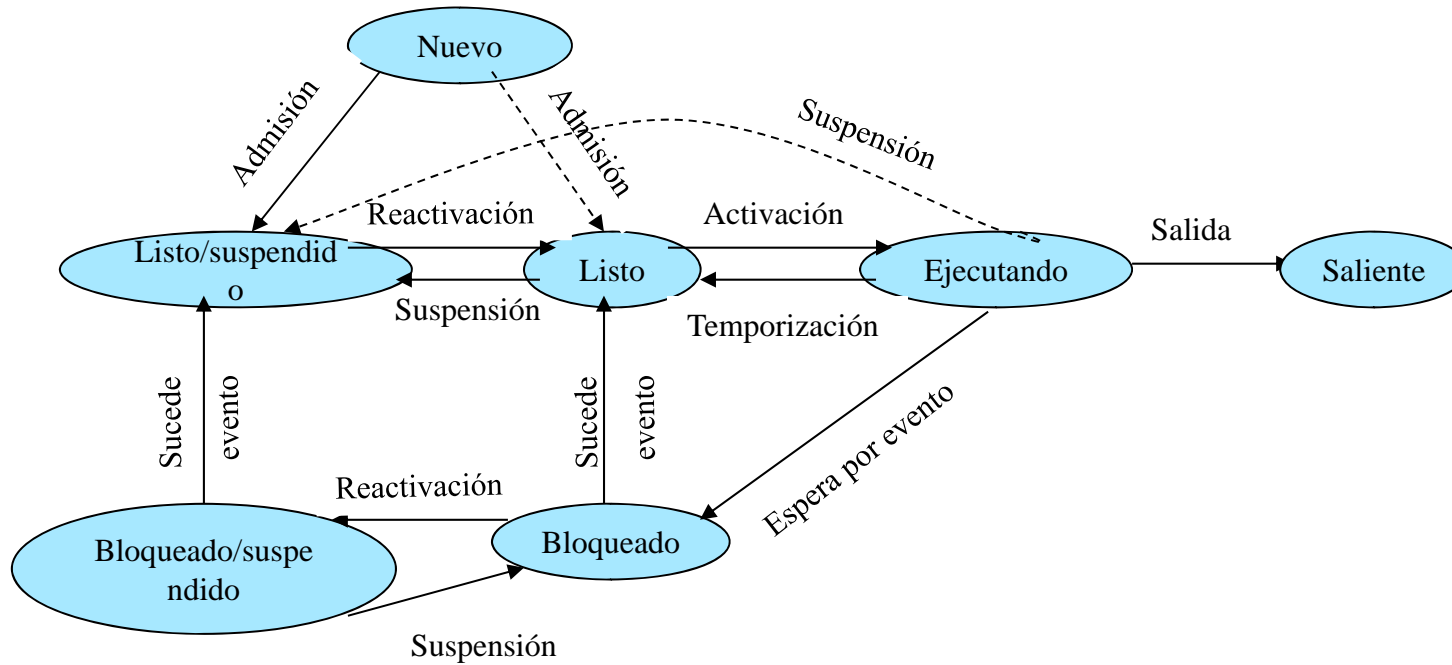
15



(a) Con un único estado Suspendido

# Dos estados suspendidos

16



(b) Con dos estados Suspendidos

Diagrama de transición de estados de procesos con estado suspendidos



# Razones para la suspensión de un proceso

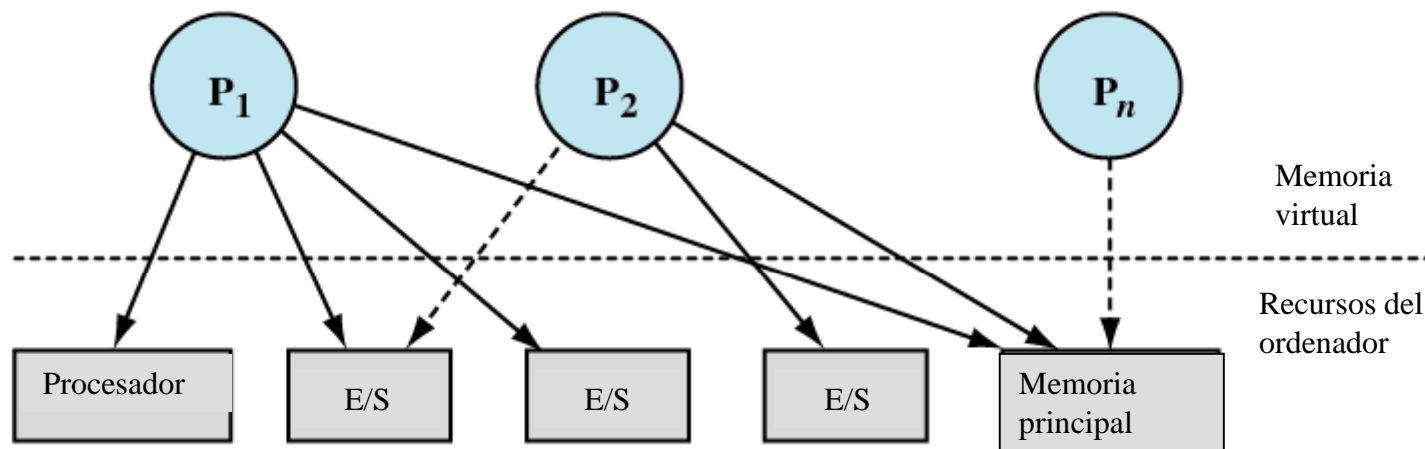
17

**Tabla 3.3. Razones para la suspensión de un proceso**

<i>Swapping</i>	El sistema operativo necesita liberar suficiente memoria principal para traer un proceso en estado Listo de ejecución.
<b>Otras razones del sistema operativo</b>	El sistema operativo puede suspender un proceso en segundo plano o de utilidad o un proceso que se sospecha puede causar algún problema.
<b>Solicitud interactiva del usuario</b>	Un usuario puede desear suspender la ejecución de un programa con motivo de su depuración o porque está utilizando un recurso.
<b>Temporización</b>	Un proceso puede ejecutarse periódicamente (por ejemplo, un proceso monitor de estadísticas sobre el sistema) y puede suspenderse mientras espera el siguiente intervalo de ejecución.
<b>Solicitud del proceso padre</b>	Un proceso padre puede querer suspender la ejecución de un descendiente para examinar o modificar dicho proceso suspendido, o para coordinar la actividad de varios procesos descendientes.

# Procesos y recursos

18

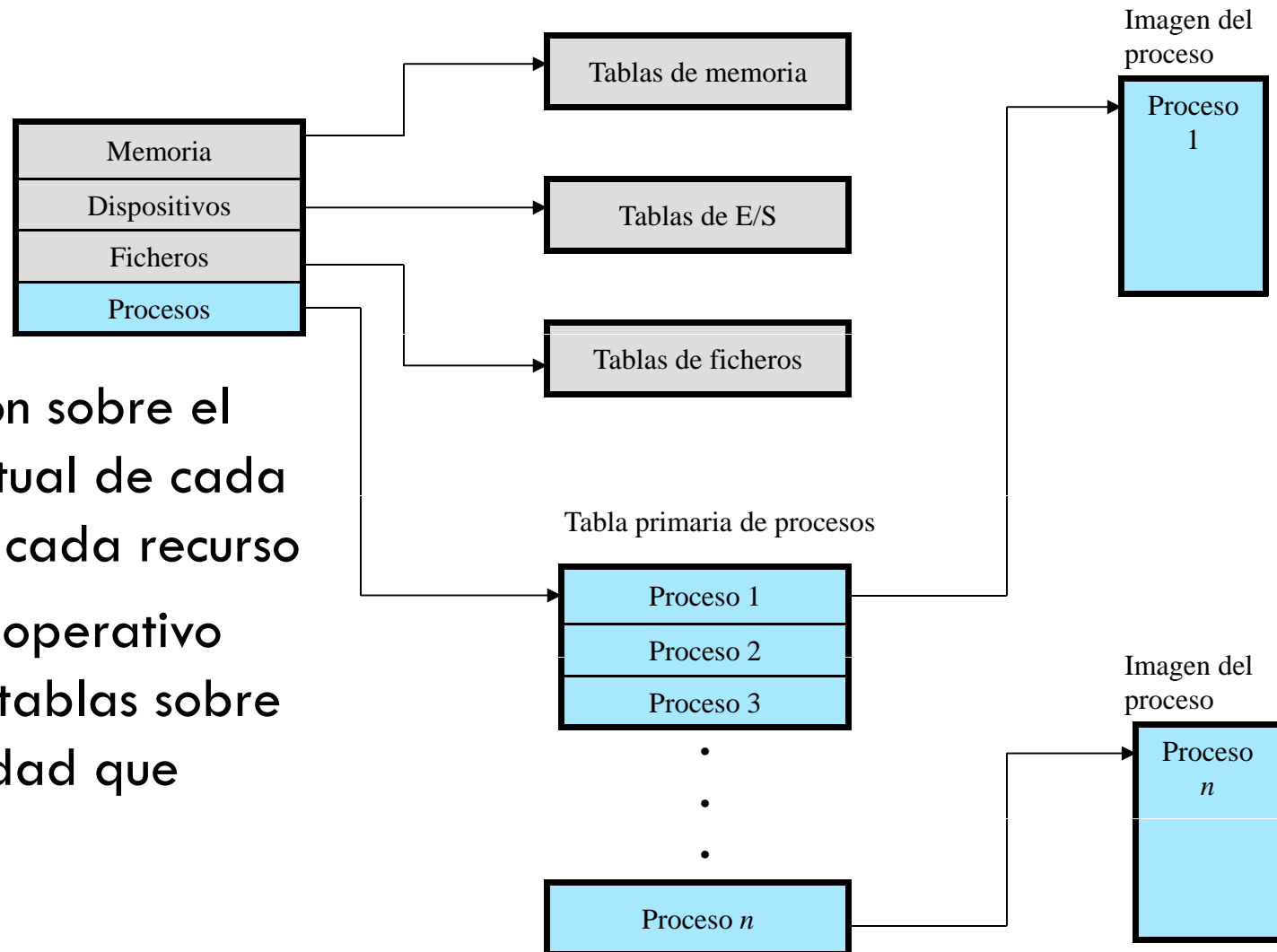


**Procesos y recursos (reserva de recursos en una instantánea del sistema)**

# Estructuras de control del sistema operativo

19

- Información sobre el estado actual de cada proceso y cada recurso
- El sistema operativo construye tablas sobre cada entidad que gestiona



# Bloque de control del proceso

20

- Identificación del proceso
- Información de estado del procesador
  - Registros visibles por el usuario
  - Registros de estado y control
  - Punteros de pila
- Información de control de proceso
  - Información de estado y de planificación
  - Comunicación entre procesos
  - Privilegios de proceso
  - Gestión de memoria
  - Propia de recursos y utilización

# Modos de ejecución

21

- Modo usuario
  - ▣ Modo menos privilegiado
  - ▣ Los programas de usuario típicamente se ejecutan en este modo
- Modo Sistema, modo control o modo núcleo
  - ▣ Modo más privilegiado
  - ▣ Núcleo del sistema operativo

# Creación de procesos

22

- Asignar un identificador de proceso único al proceso
- Reservar espacio para proceso
- Inicialización del bloque de control de proceso
- Establecer los enlaces apropiados
  - Por ejemplo: si el sistema operativo mantiene cada cola del planificador como una lista enlazada, el nuevo proceso debe situarse en la cola de Listos o en la cola de Listos/Suspendidos
- Creación o expansión de otras estructuras de datos
  - Por ejemplo: mantener un registro de auditoría

# Cuándo se realiza el cambio de proceso

23

- Interrupción de reloj
  - ▣ El proceso ha ejecutado la máxima rodaja de tiempo permitida
- Interrupción de E/S
- Fallo de memoria
  - ▣ El procesador se encuentra con una referencia a una dirección de memoria virtual, por lo que ésta debe llevarse a la memoria principal

# Cuándo se realiza el cambio de proceso

24

## □ *Trap*

- ▣ Con él, el sistema operativo conoce si una condición de error o excepción es irreversible
- ▣ Puede provocar que el proceso se mueva al estado Saliente

## □ Llamada al sistema

- ▣ Abrir un archivo



# Cambio del estado del proceso

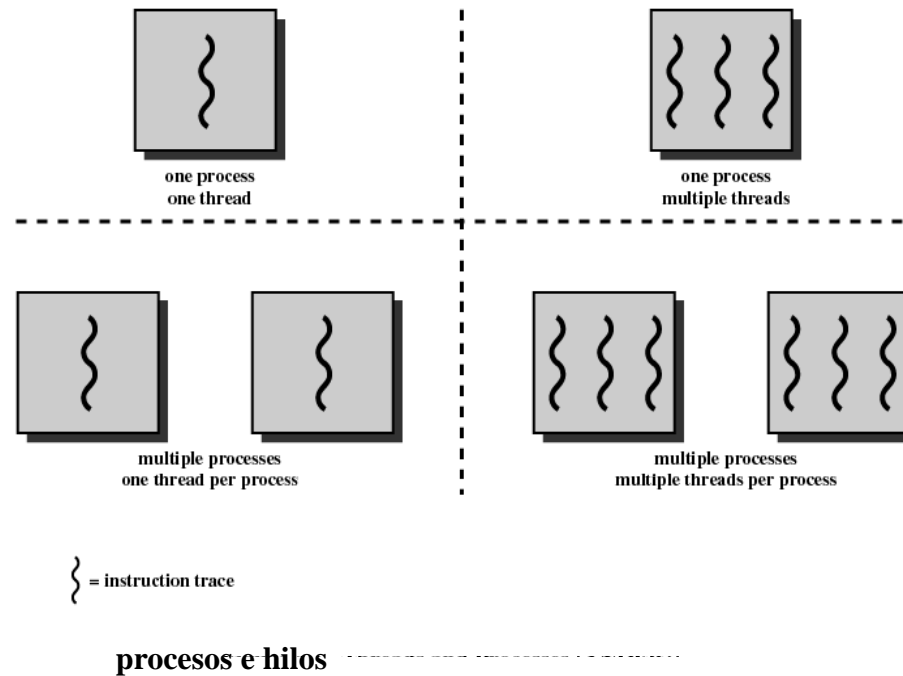
25

- ❑ Salvar el estado del procesador, incluyendo el contador de programa y otros registros
- ❑ Actualizar el bloque de control del proceso que está actualmente en el estado Ejecutando.
- ❑ Mover el bloque de control de proceso a la cola apropiada (Listo, Bloqueado, Listo/Suspendido)
- ❑ Seleccionar otro nuevo proceso para ejecutar
- ❑ Actualizar el bloque de control del proceso elegido
- ❑ Actualizar las estructuras de datos de gestión de memoria
- ❑ Restaurar el contexto del proceso seleccionado

# Hilos (threads)

26

- Flujos de control independientes dentro de un mismo proceso con pila, variables locales y CP propios



# Hilos (threads)/Ejemplos

27

- El sistema operativo da soporte a múltiples hilos de ejecución en un solo proceso.
- El MS-DOS da soporte a un único hilo.
- UNIX da soporte a múltiples procesos de usuario pero tan sólo da soporte a un hilo por proceso.
- Windows, Solaris, Linux, Mach, and OS/2 dan soporte a múltiples hilos.

# Hilos (threads)

28

- Un estado de ejecución por hilo (Ejecutando, Listo, etc.).
- Un contexto de hilo que se almacena cuando no está en ejecución.
- Tiene una pila de ejecución.
- Por cada hilo, espacio de almacenamiento estático para variables locales.
- Acceso a la memoria y recursos de su proceso compartido con todos los hilos de su mismo proceso.

# Hilos (threads)

29

- Ventajas
  - ▣ Lleva menos tiempo crear/finalizar/conmutar hilos que procesos.
  - ▣ Comparten memoria y archivos en un proceso y se pueden comunicar entre ellos sin necesidad de invocar al núcleo.
- Inconvenientes
  - ▣ Sincronización entre los hilos y entre los procesos.
  - ▣ Complejidad en la programación y depuración
- Diseño de programas basados en hilos
  - ▣ Aplicaciones con paralelismo potencial
  - ▣ Existencia de numerosas tareas de E/S
  - ▣ Existencia de eventos asíncronos

# Planificación/objetivos

30

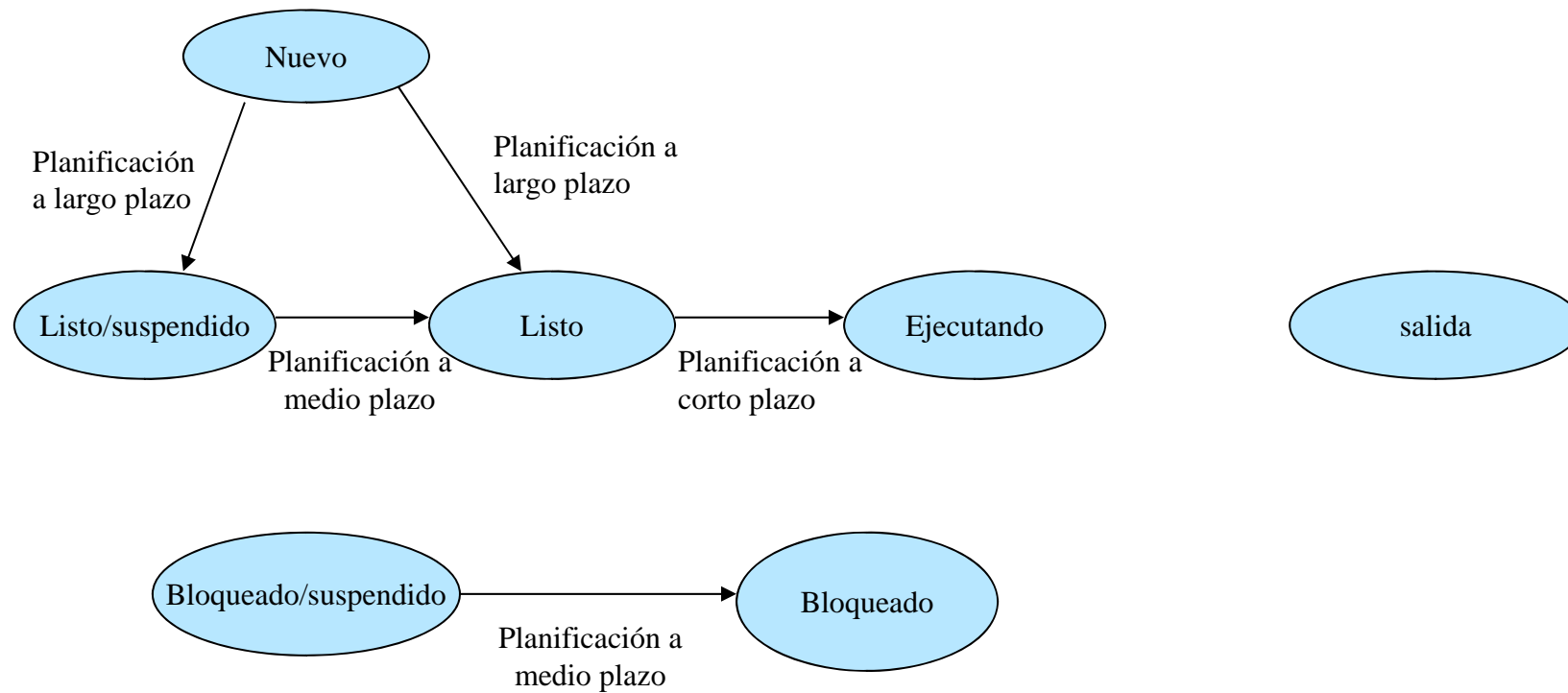
- Asignar procesos para ser ejecutados por el procesador(es)
- Tiempo de respuesta
- Rendimiento
- Eficiencia del procesador

# Planificación/tipos

31

Planificación a largo plazo	La decisión de añadir un proceso al conjunto de procesos a ser ejecutados
Planificación a medio plazo	La decisión de añadir un proceso al número de procesos que están parcialmente o totalmente en la memoria principal
Planificación a corto plazo	La decisión por la que un proceso disponible será ejecutado por el procesador
Planificación de la E/S	La decisión por la que un proceso que está pendiente de una petición de E/S será atendido por un dispositivo de E/S disponible

# Planificación/tipos

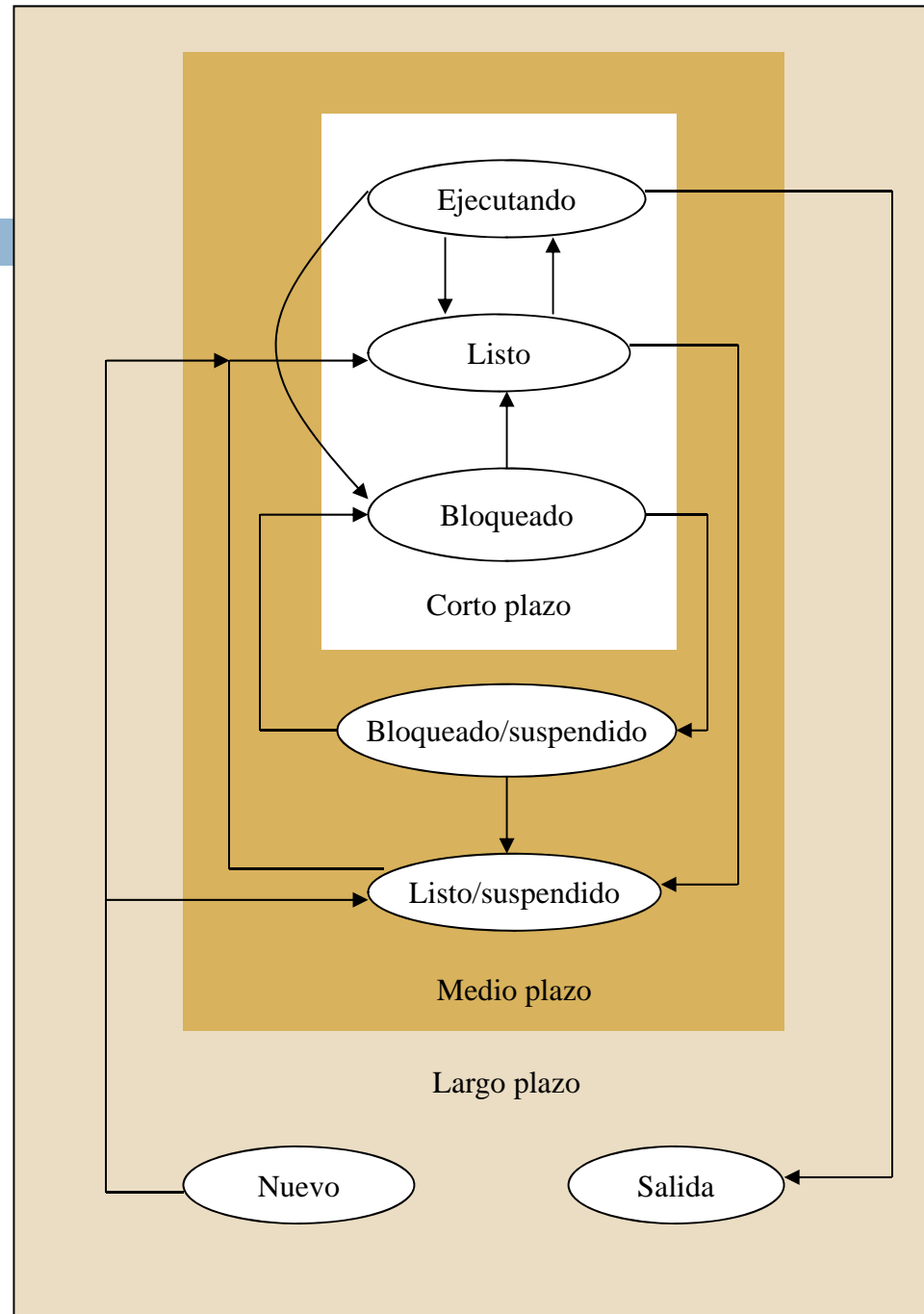


**Planificación y transiciones de estado de los procesos**



# Planificación

## /niveles



# Planificación a largo plazo

34

- Determina qué programas se admiten en el sistema para su procesamiento
- Controla el grado de multiprogramación
- Cuánto mayor sea el número de procesos creados, menor será el porcentaje de tiempo en que cada proceso se pueda ejecutar

# Planificación a medio plazo

35

- Es parte de la función de intercambio
- Se basa en la necesidad de gestionar el grado de multiprogramación

# Planificación a corto plazo

36

- Conocido como activador
- Ejecuta mucho más frecuentemente
- Se invoca siempre que ocurre un evento
  - ▣ Interrupciones de reloj
  - ▣ Interrupciones de E/S
  - ▣ Llamadas al sistema operativo
  - ▣ Señales (por ejemplo, semáforos)

# Criterios de la planificación a corto plazo

37

- Orientados al usuario
  - ▣ Tiempo de respuesta
    - Tiempo que transcurre entre el envío de una petición y la aparición de la respuesta
- Orientados al sistema
  - ▣ Uso efectivo y eficiente del procesador
- Relacionados con las prestaciones
  - ▣ Criterios cuantitativos
  - ▣ Criterios medidos como el tiempo de respuesta y el rendimiento

# Criterios

## Orientados al usuario, relacionados con las prestaciones

**Tiempo de estancia (*turnaround time*)** Tiempo transcurrido desde que se lanza un proceso hasta que finaliza. Incluye el tiempo de ejecución sumado con el tiempo de espera por los recursos, incluyendo el procesador. Es una medida apropiada para trabajos por lotes.

**Tiempo de respuesta (*response time*)** Para un proceso interactivo, es el tiempo que transcurre desde que se lanza una petición hasta que se comienza a recibir la respuesta. A menudo un proceso puede producir alguna salida al usuario mientras continúa el proceso de la petición. De esta forma, desde el punto de vista del usuario, es una medida mejor que el tiempo de estancia. La planificación debe intentar lograr bajos tiempos de respuesta y maximizar el número de usuarios interactivos con tiempos de respuesta aceptables.

**Fecha tope (*deadlines*)** Cuando se puede especificar la fecha tope de un proceso, el planificador debe subordinar otros objetivos al de maximizar el porcentaje de fechas tope conseguidas.

## Orientados al usuario, otros

**Previsibilidad** Un trabajo dado debería ejecutarse aproximadamente en el mismo tiempo y con el mismo coste a pesar de la carga del sistema. Una gran variación en el tiempo de respuesta o en el tiempo de estancia es malo desde el punto de vista de los usuarios. Puede significar una gran oscilación en la sobrecarga del sistema o la necesidad de poner a punto el sistema para eliminar las inestabilidades.

# Criterios

## Orientados al sistema, relacionados con las prestaciones

**Rendimiento** La política de planificación debería intentar maximizar el número de procesos completados por unidad de tiempo. Es una medida de cuánto trabajo está siendo realizado. Esta medida depende claramente de la longitud media de los procesos, pero está influenciada por la política de planificación, que puede afectar a la utilización.

**Utilización del procesador** Es el porcentaje de tiempo que el procesador está ocupado. Para un sistema compartido costoso, es un criterio significativo. En un sistema de un solo usuario y en otros sistemas, tales como los sistemas de tiempo real, este criterio es menos importante que algunos otros.

## Orientados al sistema, otros

**Equidad** En ausencia de orientación de los usuarios o de orientación proporcionada por otro sistema, los procesos deben ser tratados de la misma manera y ningún proceso debe sufrir inanición.

**Imposición de prioridades** Cuando se asignan prioridades a los procesos, la política del planificador debería favorecer a los procesos con prioridades más altas.

**Equilibrado de recursos** La política del planificador debería mantener ocupados los recursos del sistema. Los procesos que utilicen poco los recursos que en un determinado momento están sobreutilizados, deberían ser favorecidos. Este criterio también implica planificación a medio plazo y a largo plazo.

# Encolamiento

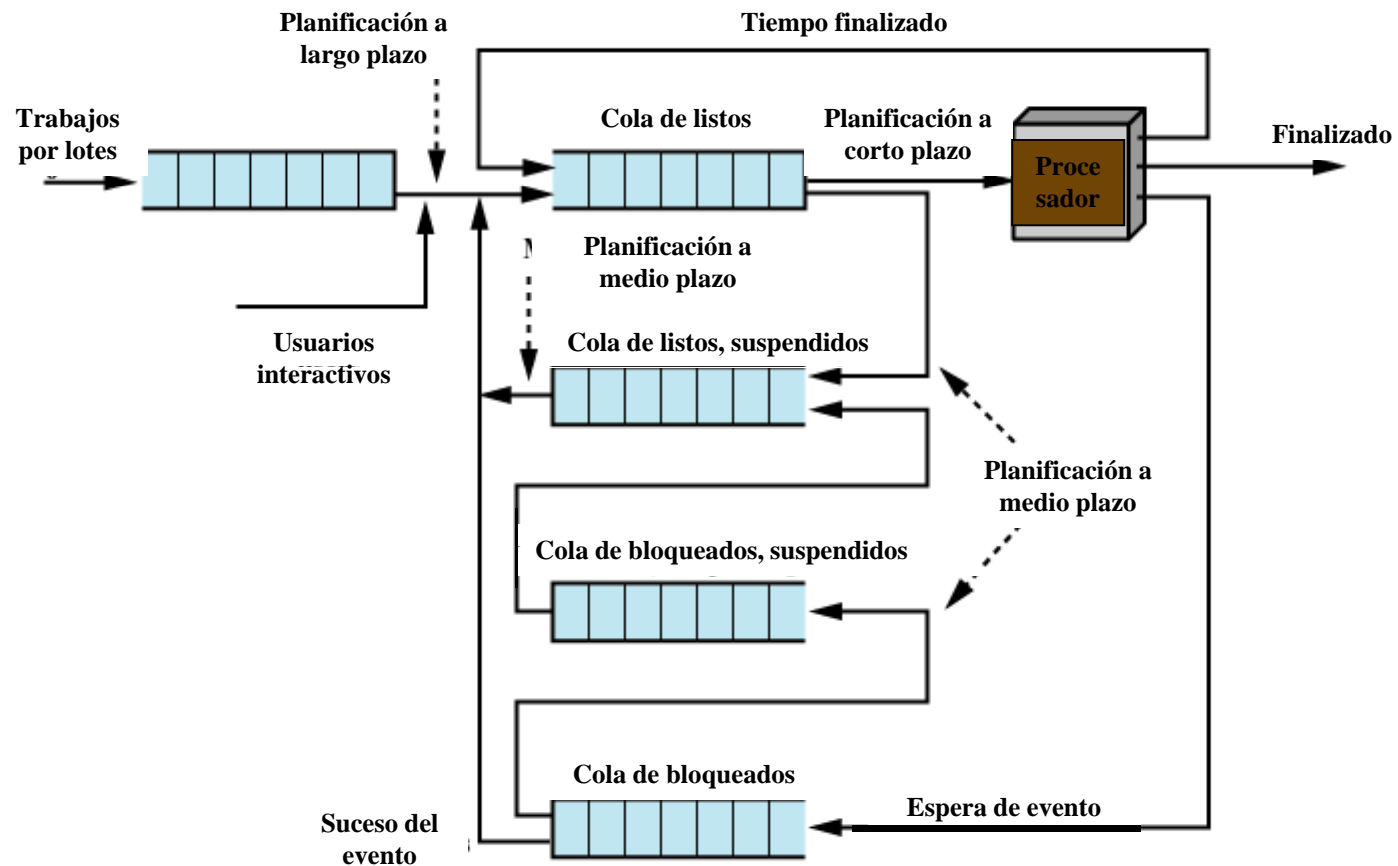


Diagrama de encolamiento para la planificación

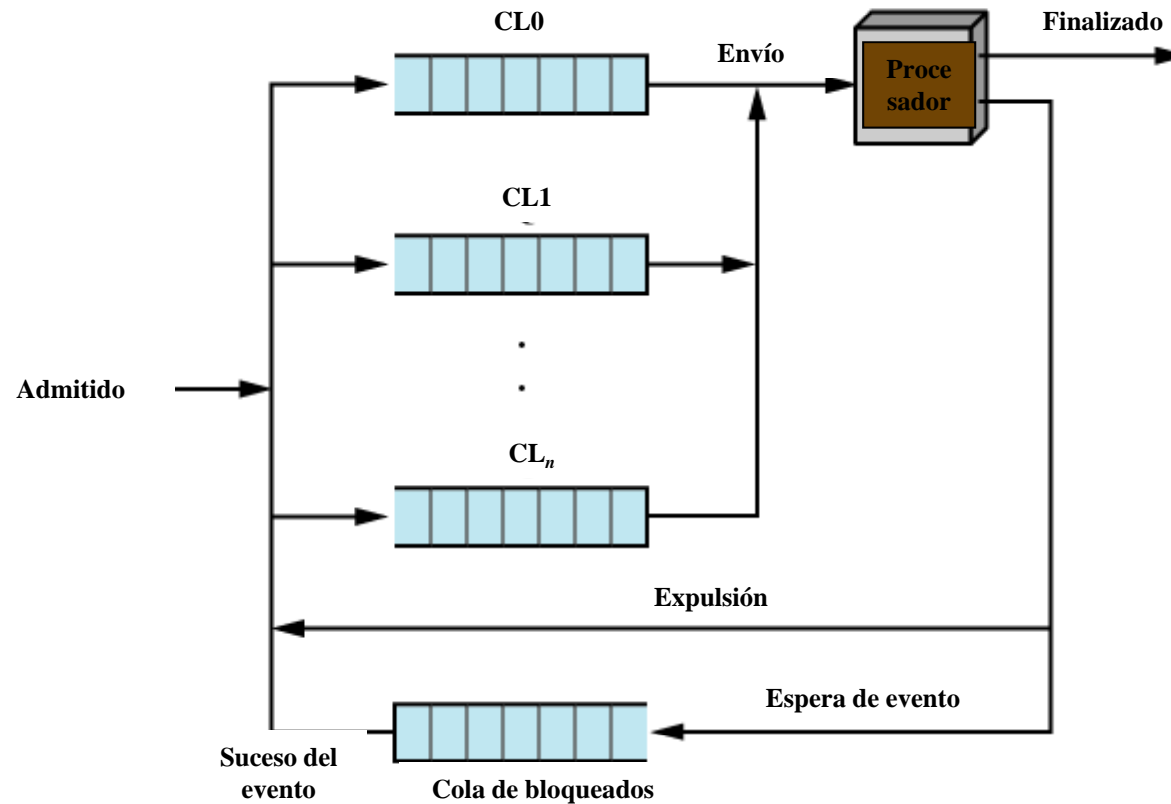


# Prioridades

41

- El planificador siempre elegirá un proceso de prioridad mayor sobre un proceso de prioridad menor
- Tiene múltiples colas de listos para representar cada nivel de prioridad
- Los procesos con prioridad más baja pueden sufrir inanición
  - ▣ Permite que un proceso cambie su prioridad basándose en su antigüedad o histórico de ejecución

# Prioridades



**Encolamiento con prioridades**

# Modo de decisión

43

- Sin expulsión (*nonpreemptive*)
  - ▣ Una vez que el proceso está en el estado Ejecutando, continuará ejecutando hasta que termina o se bloquea para esperar E/S
- Con expulsión (*preemptive*)
  - ▣ Un proceso ejecutando en un determinado momento puede ser interrumpido y pasado al estado de listo por el sistema operativo
  - ▣ Puede proporcionar mejor servicio ya que previene que cualquier proceso pueda monopolizar el procesador durante mucho tiempo

# Primero en llegar, primero en servirse [*First-Come-First-Served* (FCFS)]

Primero en llegar,  
primero en servirse  
(FCFS)

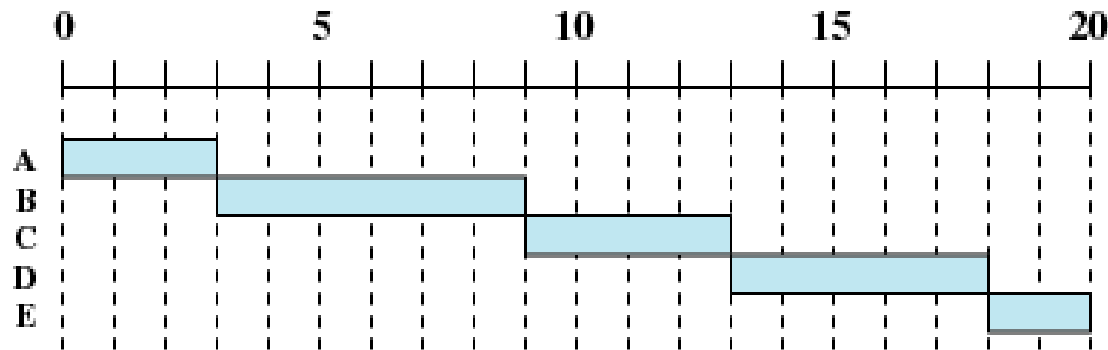


Table 9.4 Process Scheduling Example

Process	Arrival Time	Service Time
A	0	3
B	2	6
C	4	4
D	6	5
E	8	2

- Cada proceso se une a la cola de listos
- Cuando el proceso actualmente en ejecución deja de ejecutar, se selecciona el proceso que ha estado más tiempo en la cola de listos

# Primero en llegar, primero en servirse (FCFS)

45

- Un proceso corto tendrá que esperar mucho tiempo antes de poder ejecutar
- Favorece procesos limitados por el procesador
  - ▣ Los procesos limitados por la E/S deben esperar hasta que el proceso limitado por el procesador haya terminado

# Turno rotatorio [*Round Robin* (RR)]

Turno rotatorio  
(RR),  $q = 1$

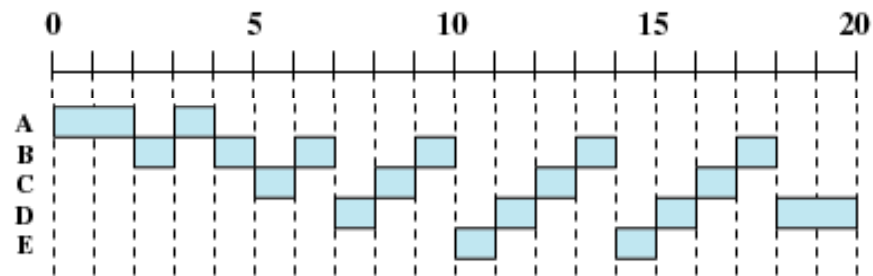


Table 9.4 Process Scheduling Example

Process	Arrival Time	Service Time
A	0	3
B	2	6
C	4	4
D	6	5
E	8	2

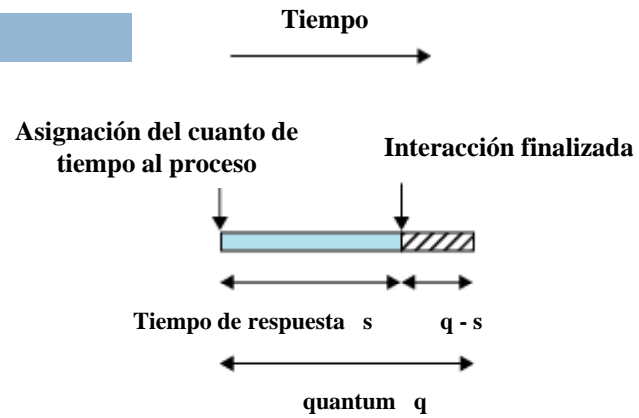
- Utiliza la expulsión basándose en el reloj
- Se determina una cantidad de tiempo, lo que permite que cada proceso utilice el procesador durante ese intervalo

# Turno rotatorio (*Round Robin*)

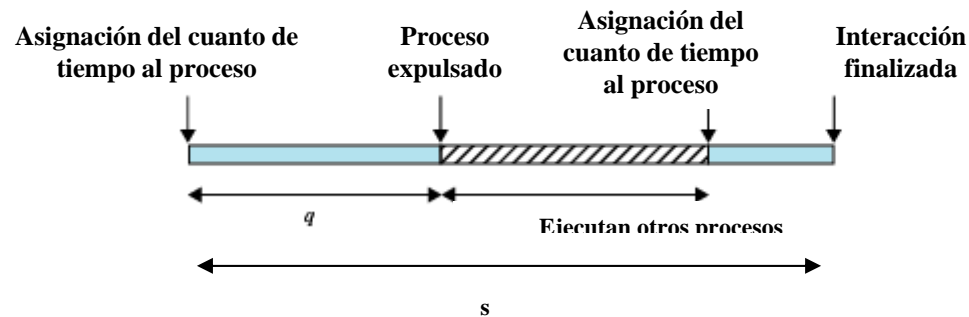
47

- Las interrupciones de reloj se suceden cada cierto intervalo de tiempo
- Cuando sucede la interrupción, el proceso actual en ejecución se sitúa en la cola de listos
  - ▣ Se selecciona el siguiente trabajo
- Esta técnica se conoce como cortar el tiempo (*time slicing*)

# Turno rotatorio (*Round Robin*)



(a) Cuanto de tiempo mayor que la interacción típica



(b) Cuanto de tiempo menor que la interacción típica

Efecto del tamaño del *quantum* de tiempo de expulsión



# Turno rotatorio (*Round Robin*)

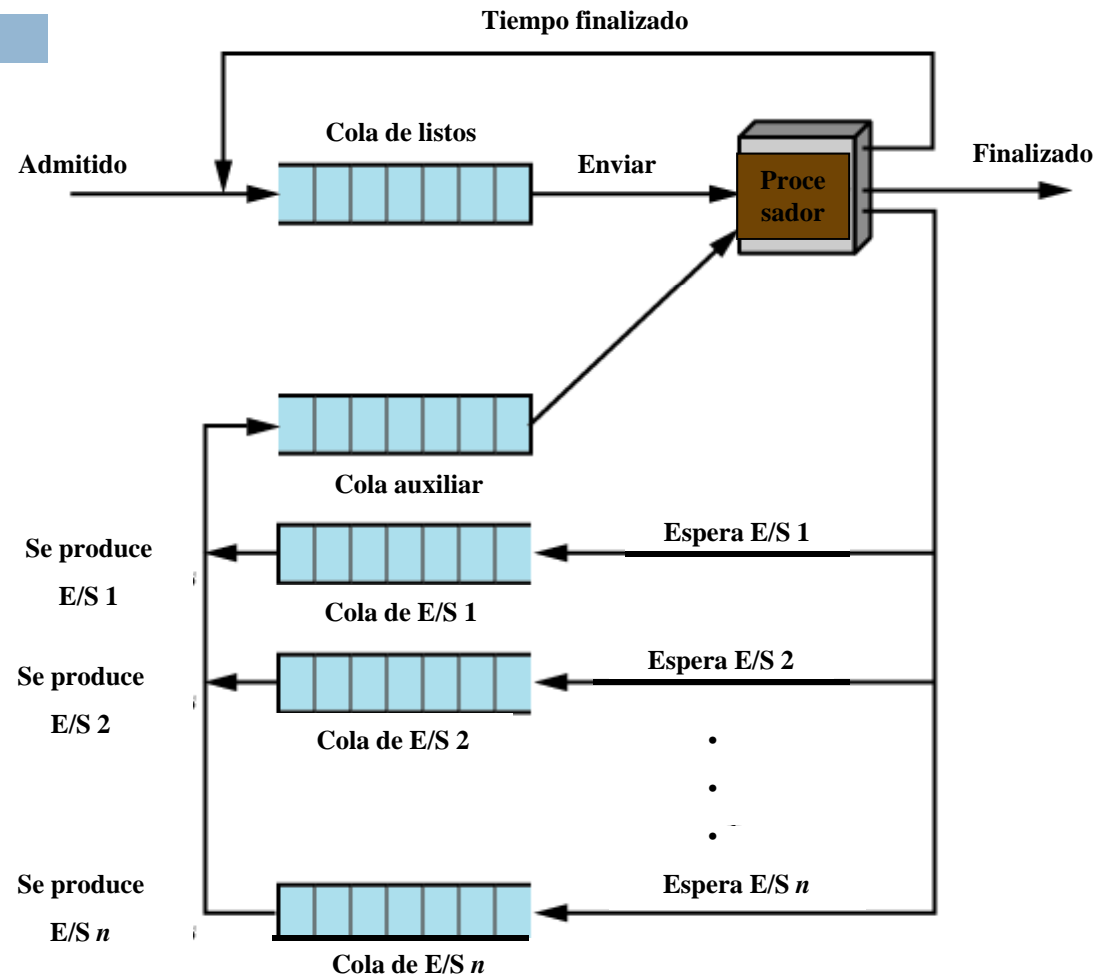


Diagrama de encolamiento para el planificador en turno rotatorio virtual

# Primero el proceso más corto [*Shortest Process Next (SPN)*]

Primero el proceso  
más corto (SPN)

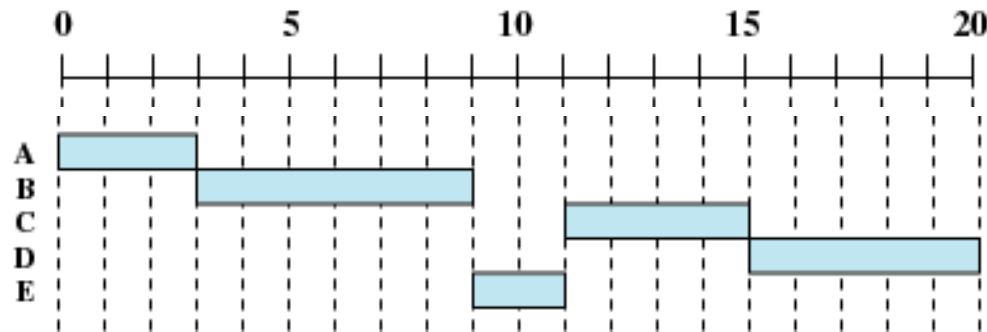


Table 9.4 Process Scheduling Example

Process	Arrival Time	Service Time
A	0	3
B	2	6
C	4	4
D	6	5
E	8	2

- Política no expulsiva
- Se selecciona el proceso con el tiempo de procesamiento esperado más corto
- El proceso más corto se situará a la cabeza de la cola, delante de los procesos más largos

# Primero el proceso más corto

## *(Shortest Process Next)*

51

- Se reduce la predecibilidad de los procesos más largos
- Si el tiempo estimado para un proceso no es el correcto, el sistema operativo podría abortar el trabajo
- Posibilidad de inanición para los procesos más largos

# Menor tiempo restante [*Shortest Remaining Time (SRT)*]

Menor tiempo  
restante (SRT)

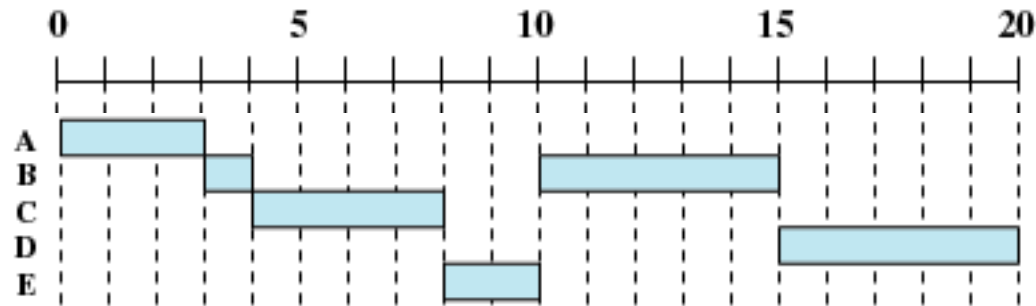


Table 9.4 Process Scheduling Example

Process	Arrival Time	Service Time
A	0	3
B	2	6
C	4	4
D	6	5
E	8	2

- La política del menor tiempo restante es una versión expulsiva de primero el proceso más corto (SPN)
- Debe tener una estimación del tiempo de proceso

# Retroalimentación (*feedback*)

Retroalimentación

$q = 1$

Retroalimentación

$q = 2^i$

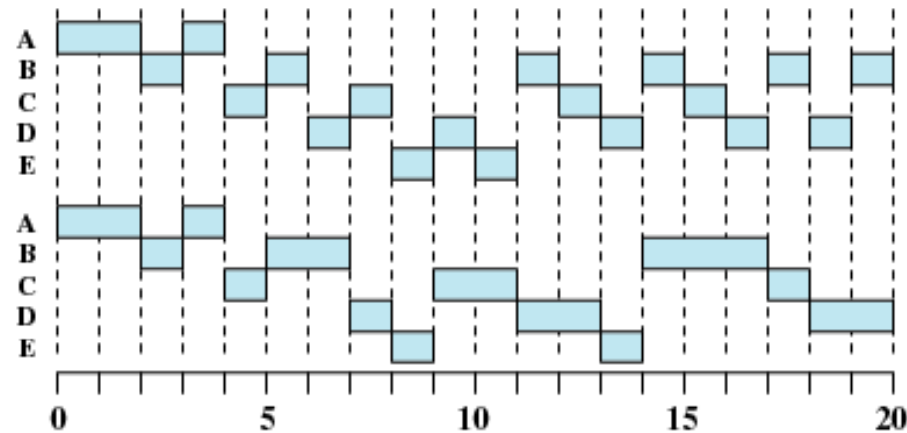
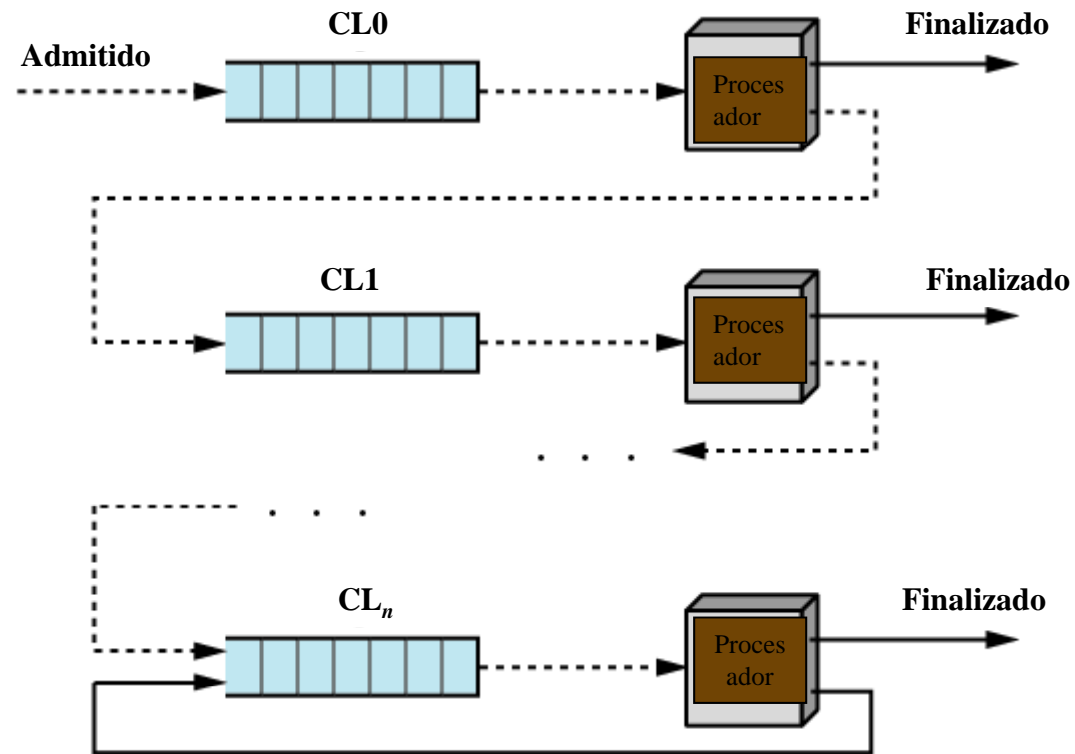


Table 9.4 Process Scheduling Example

Process	Arrival Time	Service Time
A	0	3
B	2	6
C	4	4
D	6	5
E	8	2

- Se penaliza a los trabajos que han estado ejecutando más tiempo
- Si no podemos basarnos en el tiempo de ejecución restante, nos podemos basar en el tiempo de ejecución utilizado hasta el momento
- Los procesos van descendiendo a colas inferiores después de cada ejecución
- Variante: Puedes ejecutar  $2^q$  unidades de tiempo.

# Retroalimentación (*feedback*)



**Planificación retroalimentada**

	Función de Selección	Modo de Decisión	Rendimiento	Tiempo de respuesta	Rendimiento	Efecto sobre los Procesos	Inanición
<b>FCFS</b>	$\max[w]$	No expulsiva	No especificado	Puede ser alto especialmente si hay mucha diferencia entre los tiempos de ejecución de los procesos	Mínimo	Penaliza procesos cortos; penaliza procesos con mucha E/S	No
<b>Turno Rotatorio</b> ( <i>round robin</i> )	constante	Expulsiva (por rodajas de tiempo)	Puede ser mucho si la rodaja es demasiado pequeña	Proporciona buen tiempo de respuesta para procesos cortos	Mínimo	Tratamiento justo	No
<b>SPN</b>	$\min[s]$	No expulsiva	Alto	Proporciona buen tiempo de respuesta para procesos cortos	Puede ser alto	Penaliza procesos largos	Posible
<b>SRT</b>	$\min[s-e]$	Expulsiva (a la llegada)	Alto	Proporciona buen tiempo de respuesta	Puede ser alto	Penaliza procesos largos	Posible
<b>HRRN</b>	$\max(w+s/s)$	No expulsiva	Alto	Proporciona buen tiempo de respuesta	Puede ser alto	Buen equilibrio	No
<b>Feedback</b>	(ver texto)	Expulsiva (por rodajas de tiempo)	No especificado	No especificado	Puede ser alto	Puede favorecer procesos con mucha E/S	Posible

$w$  = tiempo de espera

$e$  = tiempo de ejecución hasta el momento

$s$  = tiempo total de servicio requerido por el proceso, incluyendo  $e$

**Características de algunas políticas de planificación**

## Comparación de las políticas de planificación

	Proceso	A	B	C	D	E	Mean
	Tiempo de llegada	0	2	4	6	8	
	Tiempo de servicio ( $T_s$ )	3	6	4	5	2	
FCFS	Tiempo de finalización	3	9	13	18	20	8.60
	Tiempo de estancia ( $Tr$ )	3	7	9	12	12	
	$Tr/T_s$	1.00	1.17	2.25	2.40	6.00	
RR $q = 1$	Tiempo de finalización	4	18	17	20	15	10.80
	Tiempo de estancia ( $Tr$ )	4	16	13	14	7	
	$Tr/T_s$	1.33	2.67	3.25	2.80	3.50	
RR $q = 4$	Tiempo de finalización	3	17	11	20	19	10.00
	Tiempo de estancia ( $Tr$ )	3	15	7	14	11	
	$Tr/T_s$	1.00	2.5	1.75	2.80	5.50	
SPN	Tiempo de finalización	3	9	15	20	11	7.60
	Tiempo de estancia ( $Tr$ )	3	7	11	14	3	
	$Tr/T_s$	1.00	1.17	2.75	2.80	1.50	
SRT	Tiempo de finalización	3	15	8	20	10	7.20
	Tiempo de estancia ( $Tr$ )	3	13	4	14	2	
	$Tr/T_s$	1.00	2.17	1.00	2.80	1.00	
HRRN	Tiempo de finalización	3	9	13	20	15	8.00
	Tiempo de estancia ( $Tr$ )	3	7	9	14	7	
	$Tr/T_s$	1.00	1.17	2.25	2.80	3.5	
FB $q = 1$	Tiempo de finalización	4	20	16	19	11	10.00
	Tiempo de estancia ( $Tr$ )	4	18	12	13	3	
	$Tr/T_s$	1.33	3.00	3.00	2.60	1.5	
FB $q = 2^i$	Tiempo de finalización	4	17	18	20	14	10.60
	Tiempo de estancia ( $Tr$ )	4	15	14	14	6	
	$Tr/T_s$	1.33	2.50	3.50	2.80	3.00	



# Aspectos Multimedia

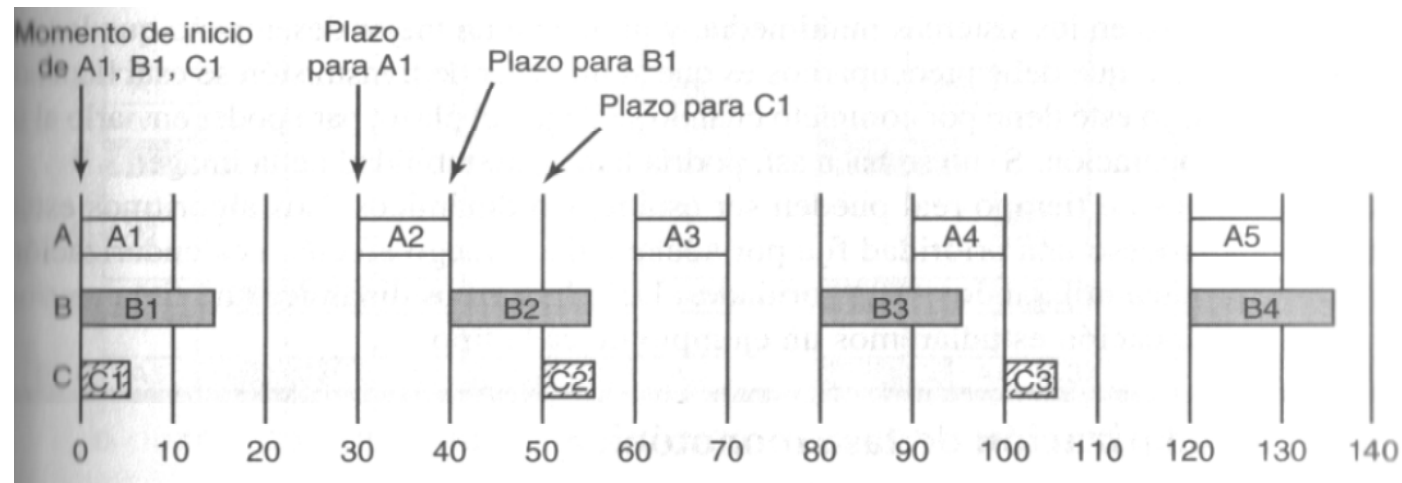
57

- Los sistemas operativos que manejan multimedia difieren de los tradicionales en:
  - ▣ Planificación de procesos
  - ▣ Sistema de archivos
  - ▣ Planificación de disco

# Planificación Multimedia

58

- Planificación de proceso homogéneos
  - ▣ Servidores de video con número fijo de películas, con mismo frame-rate y definición
  - ▣ Se usaría round-robin asegurando la frecuencia correcta con un reloj basado en esta. Ej. 30/seg NTSC
  - ▣ No es factible en la realidad.



# Planificación Multimedia

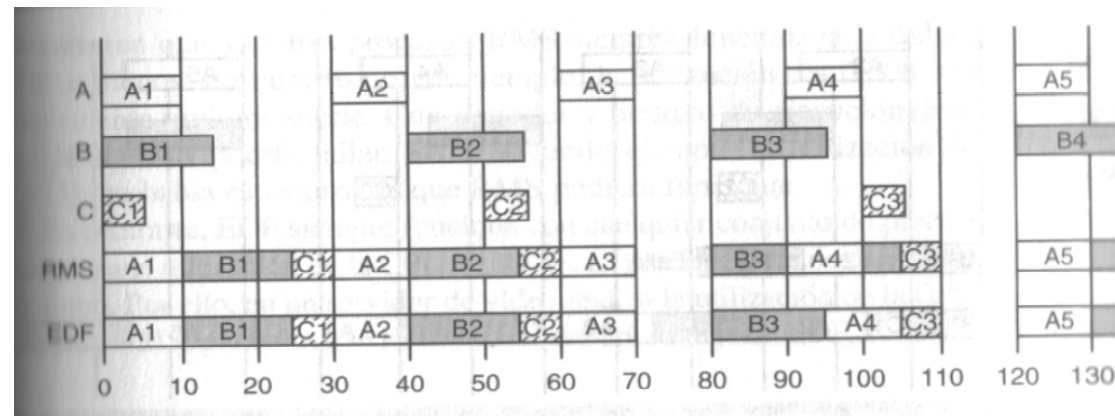
59

- Planificación de tiempo real:
  - ▣ Número de usuarios variable
  - ▣ Diferente frame-rate y formatos de compresión
  - ▣ No siempre es factible
- Enfoques
  - ▣ Planificación de tasa monotónica
  - ▣ Planificación de plazo más cercano primero

# Planificación Multimedia

60

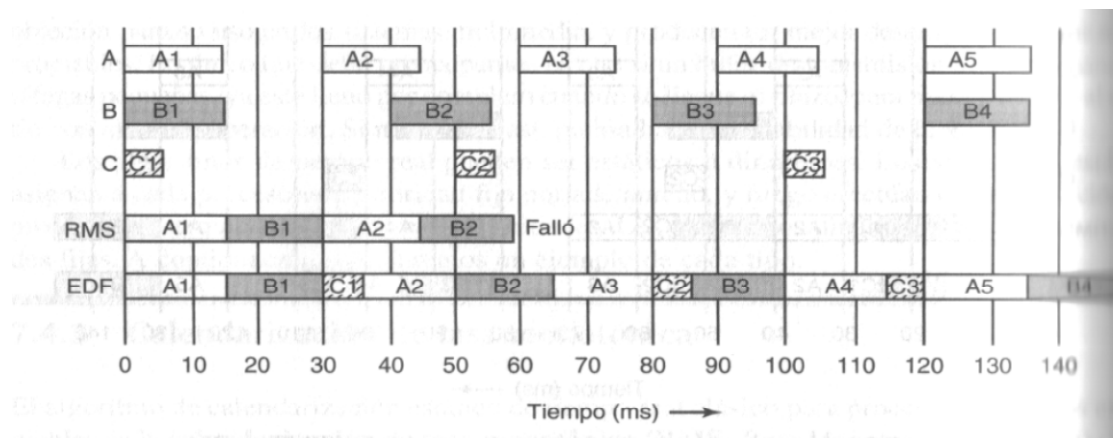
- Planificación de tasa monotónica (RMS)
- Debe cumplir:
  - ▣ Terminan en su periodo
  - ▣ No hay dependencias
  - ▣ Mismo tiempo CPU en cada ráfaga
  - ▣ Si hay no periódicos, no tienen plazo
  - ▣ Expropiación instantánea (poco factible)



# Planificación Multimedia

61

- Planificación de plazo más cercano primero (EDF)
  - ▣ Dinámico, no requiere periodicidad
  - ▣ Mantiene lista con orden de plazo
  - ▣ Expropia si plazos de procesos en lista cumplen



# Sistemas operativos de tiempo real

62

- La corrección del sistema depende no sólo del resultado lógico de la computación sino también del momento en el que se producen los resultados
- Las tareas o procesos intentan controlar o reaccionar a eventos que tienen lugar en el mundo exterior
- Estos eventos ocurren en «tiempo real» y las tareas deben ser capaces de mantener el ritmo de aquellos eventos que les conciernen

# Sistemas operativos de tiempo real

63

- Control de experimentos de laboratorio
- Control de procesos en plantas industriales
- Robótica
- Control del tráfico aéreo
- Telecomunicaciones
- Sistemas militares de mando y control

# Características de los sistemas operativos de tiempo real

64

## □ Determinista

- ▣ Realiza las operaciones en instantes de tiempo fijos predeterminados o dentro de intervalos de tiempo predeterminados
- ▣ Se preocupa de cuánto tiempo tarda el sistema operativo antes del reconocimiento de una interrupción y de si el sistema tiene capacidad suficiente para manejar todas las solicitudes dentro del tiempo requerido



# Características de los sistemas operativos de tiempo real

65

## □ Reactividad

- ▣ Se preocupa de cuánto tiempo tarda el sistema operativo, después del reconocimiento, en servir la interrupción
- ▣ Incluye la cantidad de tiempo necesario para comenzar a ejecutar la interrupción
- ▣ Incluye la cantidad de tiempo para realizar la interrupción
- ▣ El efecto del anidamiento de interrupciones

# Características de los sistemas operativos de tiempo real

66

- Control del usuario
  - ▣ El usuario especifica prioridad
  - ▣ Especifica la paginación
  - ▣ Especifica qué procesos deben residir siempre en memoria principal
  - ▣ Especifica qué algoritmos de transferencia a disco deben utilizarse
  - ▣ Especifica qué derechos tienen los procesos

# Características de los sistemas operativos de tiempo real

67

- Fiabilidad

- ▣ La degradación de sus prestaciones puede tener consecuencias catastróficas

- Operación de fallo suave

- ▣ Habilidad del sistema de fallar de tal manera que se preserve tanta capacidad y datos como sea posible
  - ▣ Estabilidad

# Características de los sistemas operativos de tiempo real

68

- Cambio de proceso o hilo rápido
- Pequeño tamaño
- Capacidad para responder rápidamente a interrupciones externas
- Multitarea con herramientas para la comunicación entre procesos como semáforos, señales y eventos

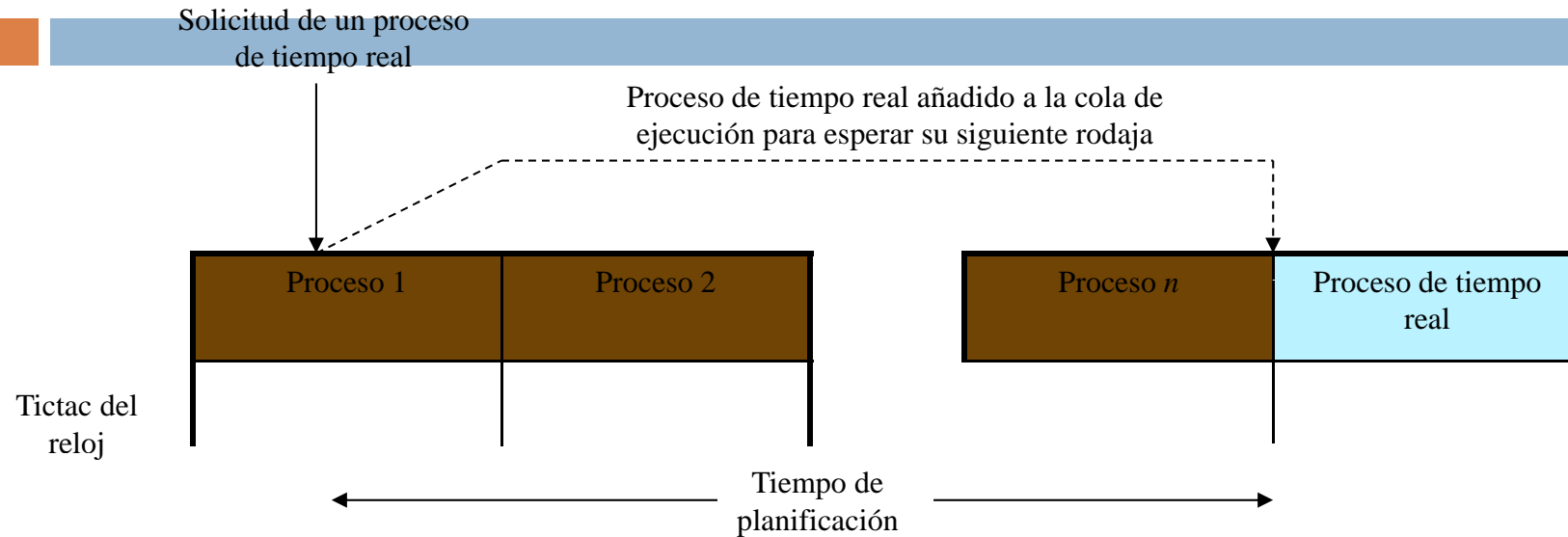
# Características de los sistemas operativos de tiempo real

69

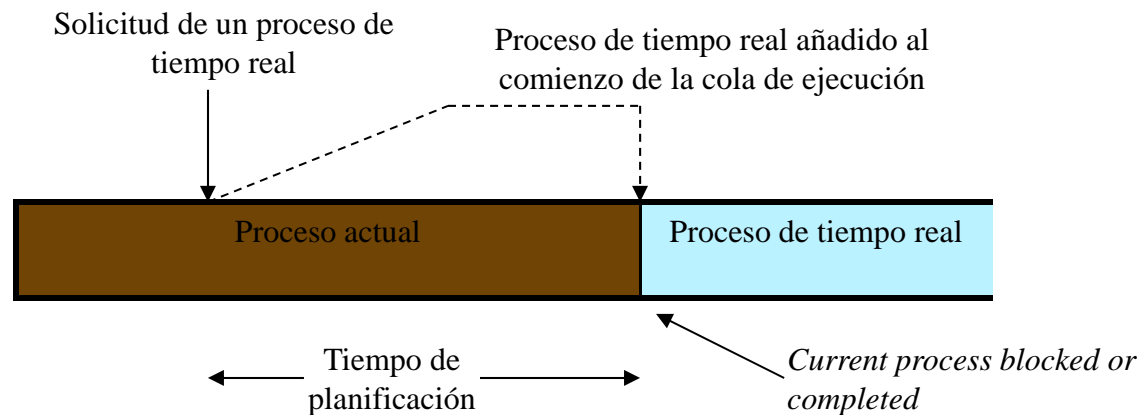
- Utilización de ficheros secuenciales especiales que pueden acumular datos a alta velocidad
- Planificación expulsiva basada en prioridades
- Minimización de los intervalos durante los cuales se deshabilitan las interrupciones
- Primitivas para retardar tareas durante una cantidad dada de tiempo y para parar/retomar tareas
- Alarmas y temporizaciones especiales

# Planificación de procesos de tiempo real

70



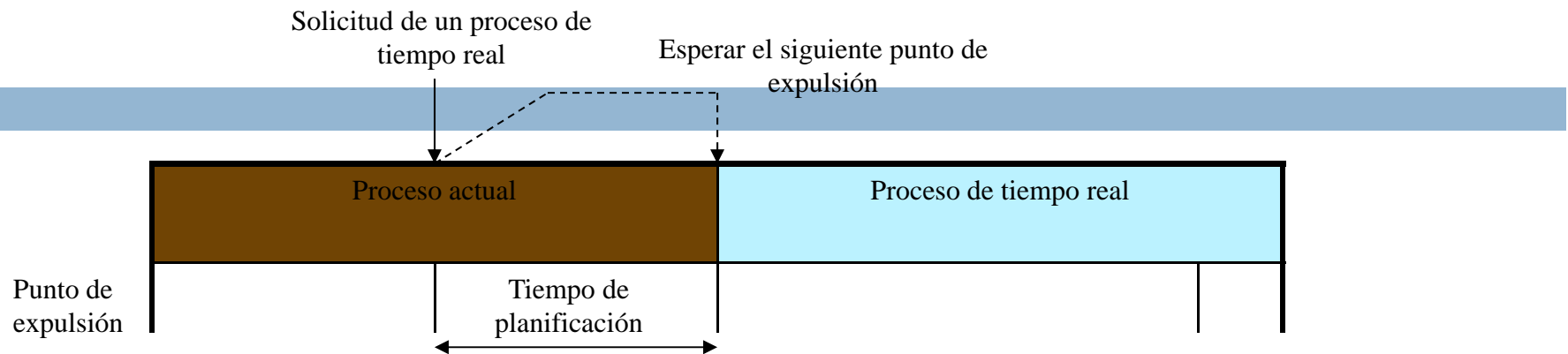
**(a) Planificador expulsivo de turno circular**



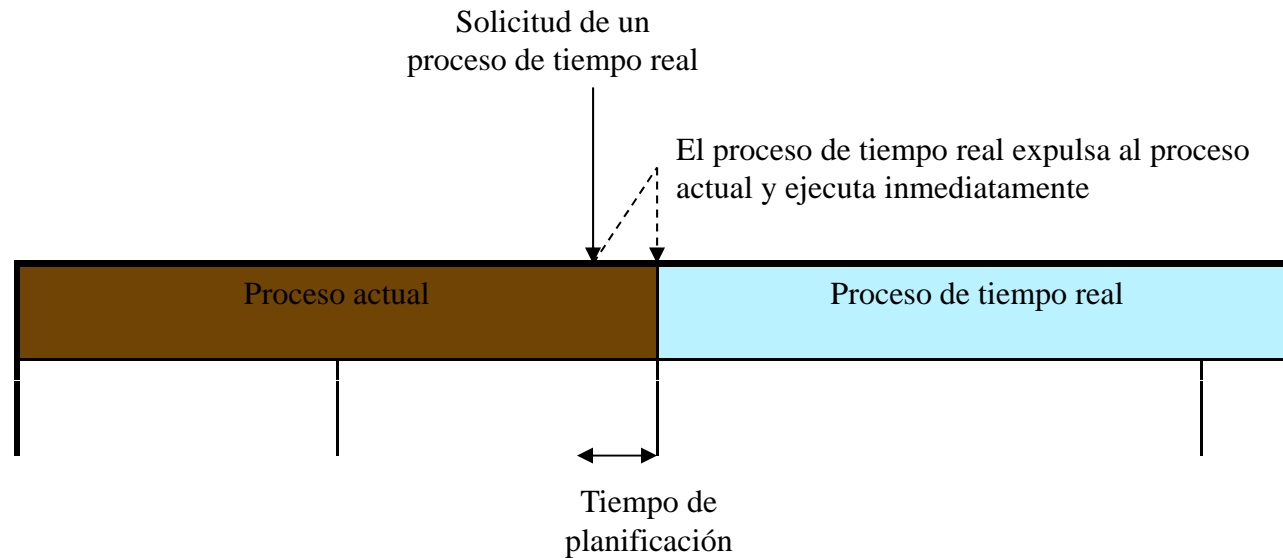
**(b) Planificador no expulsivo dirigido por prioridad**

# Planificación de procesos de tiempo real

71



**(c) Planificador expulsivo dirigido por prioridad en puntos de expulsión**



**(d) Planificador expulsivo inmediato**

**Planificación de procesos de tiempo real**

# Planificación de tiempo real

72

- Estática dirigida por tabla
  - ▣ Determina, en tiempo de ejecución, cuándo debe comenzar a ejecutarse cada tarea
- Estática con expulsión dirigida por prioridad
  - ▣ Se utiliza un planificador expulsivo tradicional basado en prioridades
- Dinámica basada en un plan
  - ▣ La factibilidad se determina en tiempo de ejecución
- Dinámica de mejor esfuerzo
  - ▣ No se realiza análisis de factibilidad



# Planificación por plazos

73

- Las aplicaciones de tiempo real no se preocupan tanto de la velocidad de ejecución como de completar sus tareas

# Planificación por plazos

74

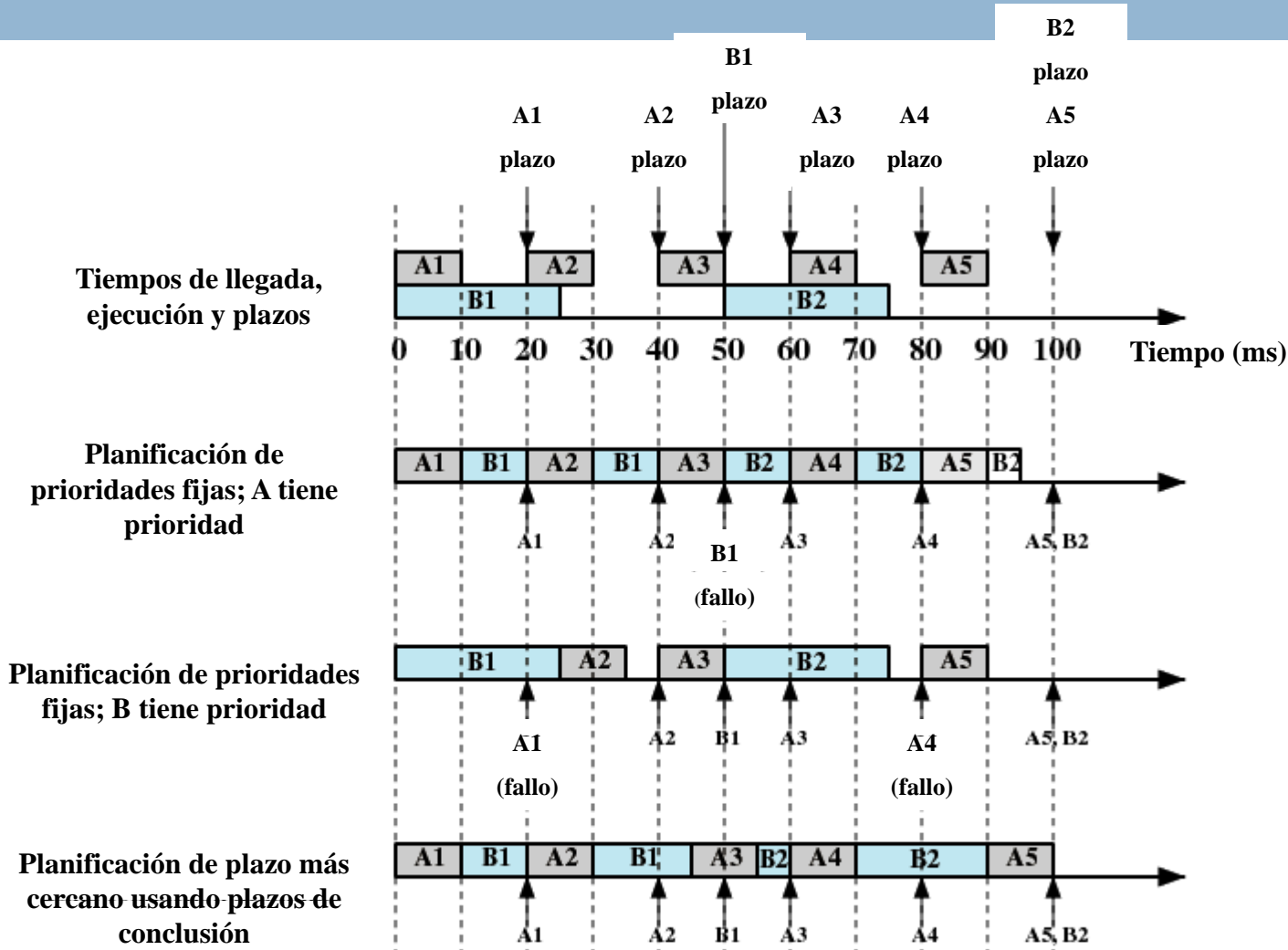
- Información utilizada
  - ▣ Tiempo de activación
  - ▣ Plazo de comienzo
  - ▣ Plazo de conclusión
  - ▣ Tiempo de proceso
  - ▣ Recursos requeridos
  - ▣ Prioridad
  - ▣ Estructura de subtareas

# Dos tareas

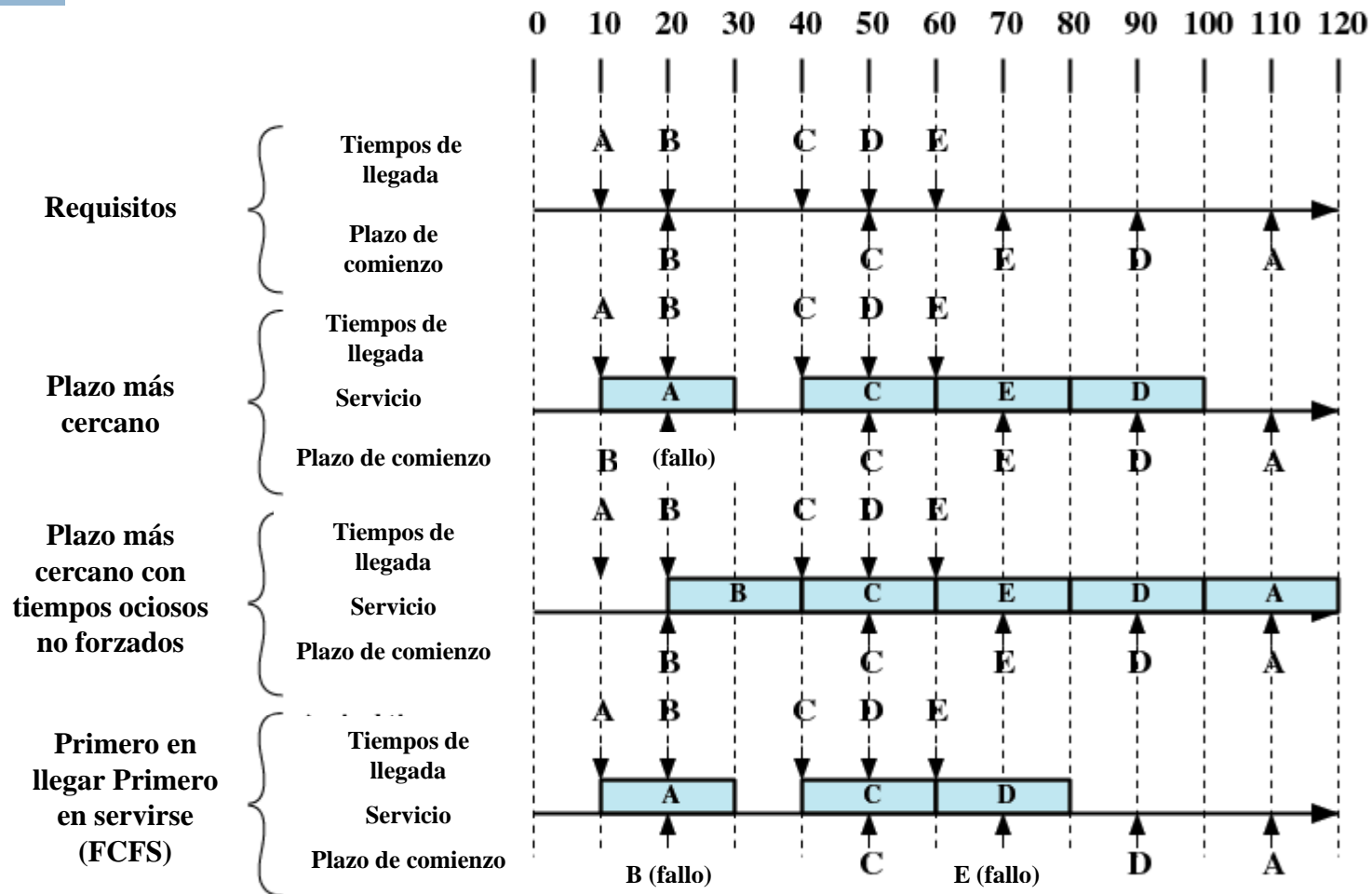
75

Perfil de ejecución de dos tareas periódicas

Proceso	Tiempo de llegada	Tiempo de ejecución	Plazo de conclusión
A(1)	0	10	20
A(2)	20	10	40
A(3)	40	10	60
A(4)	60	10	80
A(5)	80	10	100
•	•	•	•
•	•	•	•
•	•	•	•
B(1)	0	25	50
B(2)	50	25	100
•	•	•	•
•	•	•	•
•	•	•	•



Planificación de tareas periódicas de tiempo real con plazos de conclusión (basado en la Tabla 10.2)



Planificación de tareas aperiódicas de tiempo real con plazos de comienzo

## Perfil de ejecución de cinco tareas aperiódicas

Proceso	Tiempo de llegada	Tiempo de ejecución	Plazo de comienzo
A	10	20	110
B	20	20	20
C	40	20	50
D	50	20	90
E	60	20	70

# Planificación de tasa monótona

79

- Asigna prioridades a las tareas en base a sus periodos
- La tarea de mayor prioridad es aquella con el periodo más breve

# Diagrama de tiempos de una tarea periódica

80

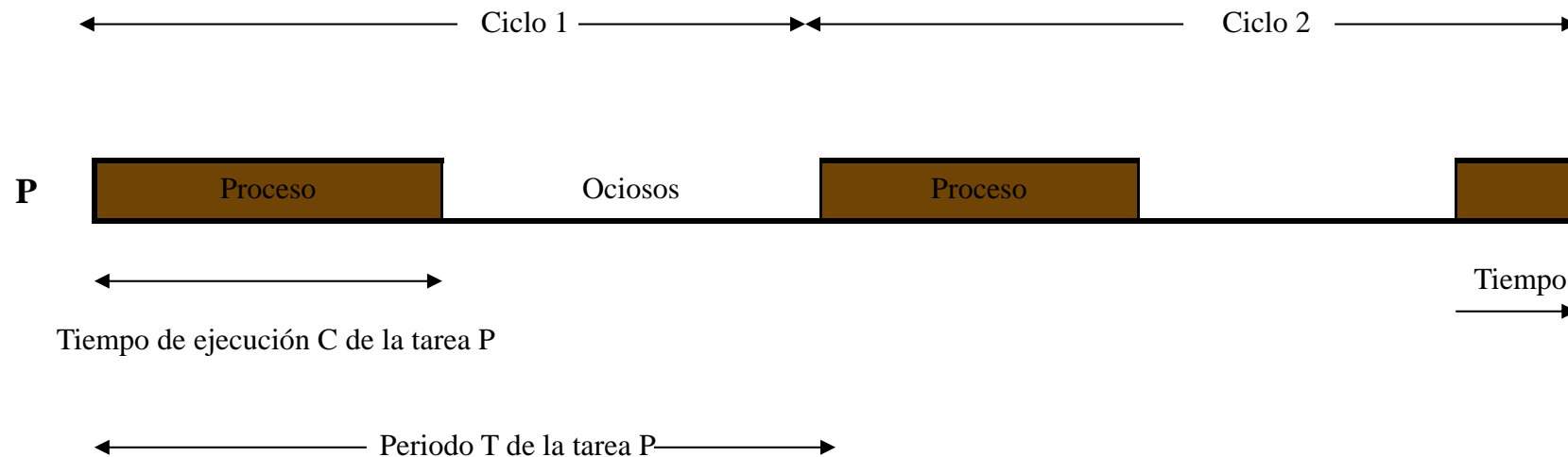


Diagrama de tiempos de una tarea periódica