

Sistemas Operativos

UNIDAD 3 – SISTEMA DE PROCESOS
PROCESOS, ESTRUCTURAS Y ESTADOS



... ya sabemos que ...

- Un sistema de computación consiste en una colección de recursos hardware como: procesador, memoria, módulos de E/S, relojes, unidades de disco, de red, etc..
- Las aplicaciones se desarrollan para realizar determinadas tareas. Aceptan entradas, realizan el trabajo y generan salidas.
- No es eficiente que el código se escriba para un hardware específico.
- Tiene sentido desarrollar código común de acceso al hardware que compartan todas las aplicaciones que accedan a ese hardware.
- El procesador por si sólo tiene un soporte limitado para la multiprogramación, necesitamos de software que gestione la compartición del procesador y otros recursos por parte de varios procesos “simultáneamente”.
- Cuando varias aplicaciones están activas hay que proteger sus datos, E/S y recursos de las demás aplicaciones.
- El S.O. proporciona una interfaz segura, consistente y completa entre las aplicaciones y el hardware. Es una capa software entre ellos.
- El S.O. Proporciona una abstracción de los recursos hardware que pueden ser solicitados por las aplicaciones y su uso controlado por el.

Los recursos deben estar disponibles para múltiples aplicaciones

El procesador debe conmutar entre múltiples aplicaciones sin dejar a ninguna sin su tiempo de proceso

El procesador, memoria y los dispositivos de E/S deben usarse con eficiencia.

Índice

- Concepto de proceso
- Estructuras de control del S.O.
 - Bloque de control de procesos
 - Tabla de procesos
 - Representación física del proceso
 - Ubicación del proceso
 - Estructura y rol del PCB
- Estados de los procesos
 - Modelo de dos estados
 - Creación y terminación de procesos
 - Modelo de cinco estados
 - Suspensión de procesos (Swapping)
 - Usos de la Suspensión

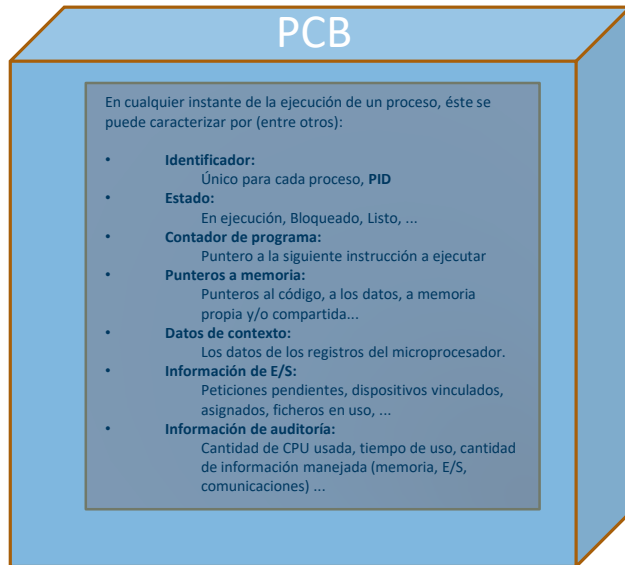
Concepto de proceso - Información en el PCB

- Un proceso es un programa en ejecución en un computador.
- Un proceso es la entidad que se puede asignar al procesador, es la entidad de ejecución.
- Un proceso es la entidad de asignación, a la que se pueden asignar los recursos (entre ellos el procesador).
- Un proceso se caracteriza por la ejecución de una secuencia de instrucciones, un estado actual y un conjunto de recursos asociados.
- Un proceso es una entidad formada por varios elementos:
 - El código del proceso
 - Los datos del proceso
 - La memoria asignada al mismo
 - La pila asociada al mismo

En cualquier instante de la ejecución de un proceso, éste se puede caracterizar por (entre otros):

- **Identificador:**
Único para cada proceso, **PID**
- **Estado:**
En ejecución, Bloqueado, Listo, ...
- **Contador de programa:**
Puntero a la siguiente instrucción a ejecutar
- **Punteros a memoria:**
Punteros al código, a los datos, a memoria propia y/o compartida...
- **Datos de contexto:**
Los datos de los registros del microprocesador.
- **Información de E/S:**
Petición pendientes, dispositivos vinculados, asignados, ficheros en uso, ...
- **Información de auditoría:**
Cantidad de CPU usada, tiempo de uso, cantidad de información manejada (memoria, E/S, comunicaciones) ...
- **Punteros de enlace**
Punteros que usa el S.O. para crear listas enlazadas entre PCBs

Concepto de proceso – Información en el PCB



Toda esta información se almacena, para cada proceso, en su **Process Control Block (PCB)**

- El S.O. crea y gestiona el PCB
- Permite restaurar el proceso tras una interrupción
- Permite dar soporte a múltiples procesos

Un proceso se caracteriza entonces por:

- Su bloque código
- Su bloque de datos
- Su memoria asignada
- Su pila de llamadas
- Su **PCB**

Estructuras de control – del Sistema Operativo

El Sistema Operativo

- Controla los eventos dentro del computador
- Planifica y activa los procesos para su ejecución
- Reserva los recursos para los procesos
- Responde a las solicitudes de servicio



Quien gestiona los recursos usados por parte de los procesos

- Está en ejecución
- Al menos parte esta en memoria principal
- Controla dos dispositivos de E/S

El Sistema Operativo gestiona todas estas asignaciones

- Al menos parte esta en memoria principal
- Está bloqueado esperando por un dispositivo E/S
- Tiene asignado un dispositivo de E/S

- Está en *Swaping*, en disco, está Suspendido
- No tiene recursos asignados
- Posiblemente esté bloqueado por otro recurso

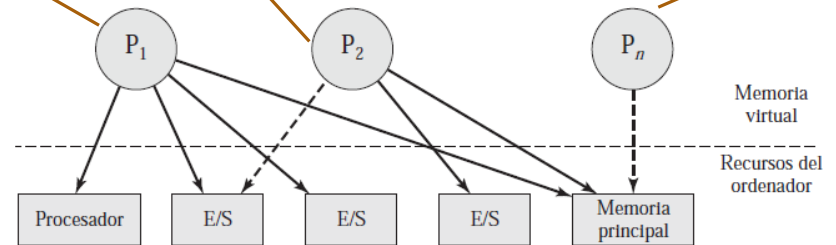


Figura 3.10. Procesos y recursos (reserva de recursos en una instantánea del sistema).

Estructuras de control – del Sistema Operativo

El Sistema Operativo:

Construye y mantiene tablas con información para cada entidad que gestiona

- Tablas de Memoria
- Tablas de Dispositivos
- Tablas de Ficheros
- Tablas de Procesos

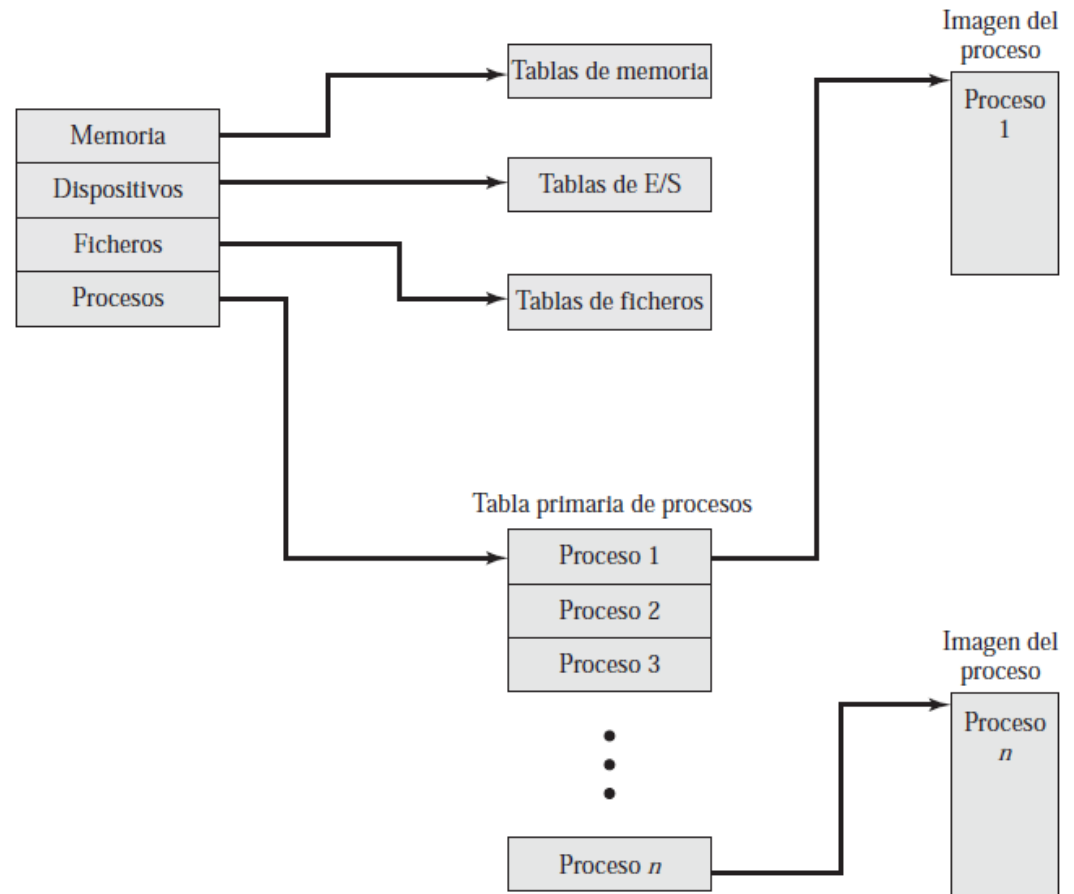


Figura 3.11. Estructura general de las tablas de control del sistema operativo.

Estructuras de control – Tablas de gestión

Tablas de Memoria

Parte de la memoria está reservada para el S.O.

Con ellas el S.O. mantiene un registro de:

- la memoria principal
- la memoria virtual

Incluyen información relativa a:

- Reservas de memoria principal por parte de qué procesos
- Reservas de memoria virtual por parte de qué procesos (Swap)
- Todos los atributos de protección, tanto principal como virtual
- Todos los atributos de compartición
- Información de ubicación y control de bloques de swap
- Reservas y asignación de memoria para los dispositivos

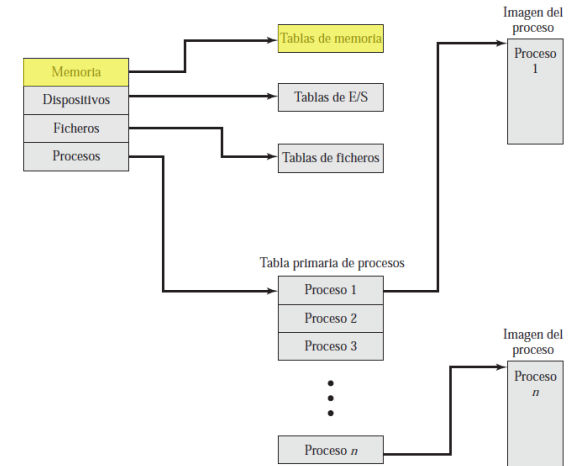


Figura 3.11. Estructura general de las tablas de control del sistema operativo.

Tablas de Dispositivos

Con ellas el S.O. gestiona:

- los dispositivos de E/S
- canales de comunicación
- canales de DMA (Direct Memory Access)

Incluyen información relativa a:

- Dispositivos/canales libres
- Asignados a qué proceso
- ¿Operación de E/S en curso? ¿estado de la operación?
- Direcciones de memoria involucradas, origen, destino.
- Tasas y volumen de las transferencias, metadatos...

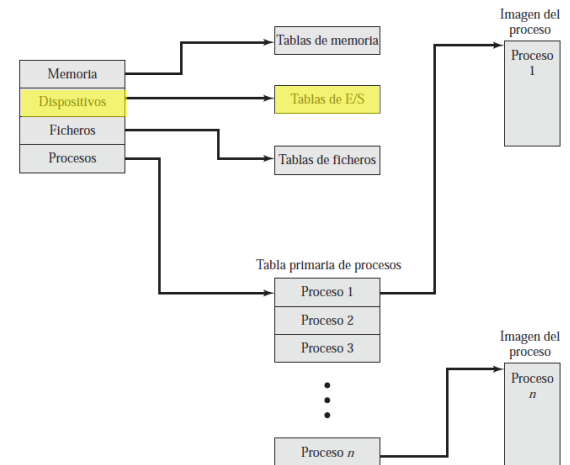


Figura 3.11. Estructura general de las tablas de control del sistema operativo.

Estructuras de control – Tablas de gestión

Tablas de Ficheros

Incluyen información relativa a:

- Existencia de ficheros
- Posición en almacenamiento secundario
- Estado actual
- Permisos

Si el S.O. delega en el sistema de ficheros, todo lo relativo a ficheros será responsabilidad del Sistema de Ficheros

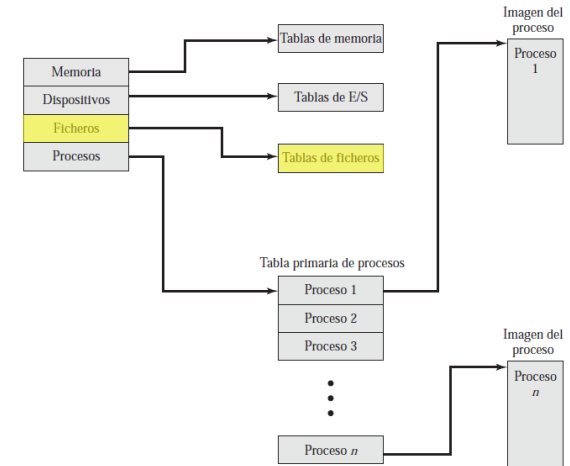


Figura 3.11. Estructura general de las tablas de control del sistema operativo.

Tablas de Procesos

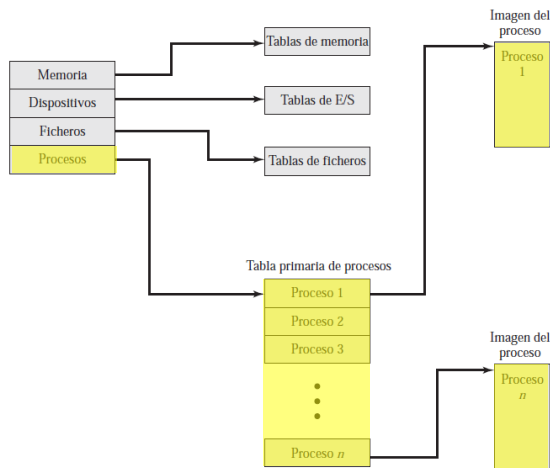


Figura 3.11. Estructura general de las tablas de control del sistema operativo.

Sobre las tablas

- ❑ Todas las tablas se encuentran entrelazadas y referenciadas entre sí mediante **referencias cruzadas**
 - Memoria, E/S, Dispositivos se gestionan para procesos → PID, punteros
 - Los ficheros residen en dispositivos → fd, device_id
 - Los procesos y ficheros pueden estar parcial/totalmente en memoria → punteros, fd, PID
- ❑ Las propias tablas residen en Memoria (principal o virtual) y son propiedad del S.O.
- ❑ El S.O. necesita información del Hardware y recursos disponibles para crearlas.
 - Cantidad de memoria física instalada
 - Qué dispositivos y canales existen y su descriptor
 - Qué capacidad de almacenamiento existe
 - etc...
- ❑ En el proceso de arranque del computador el S.O. recopila esta información.

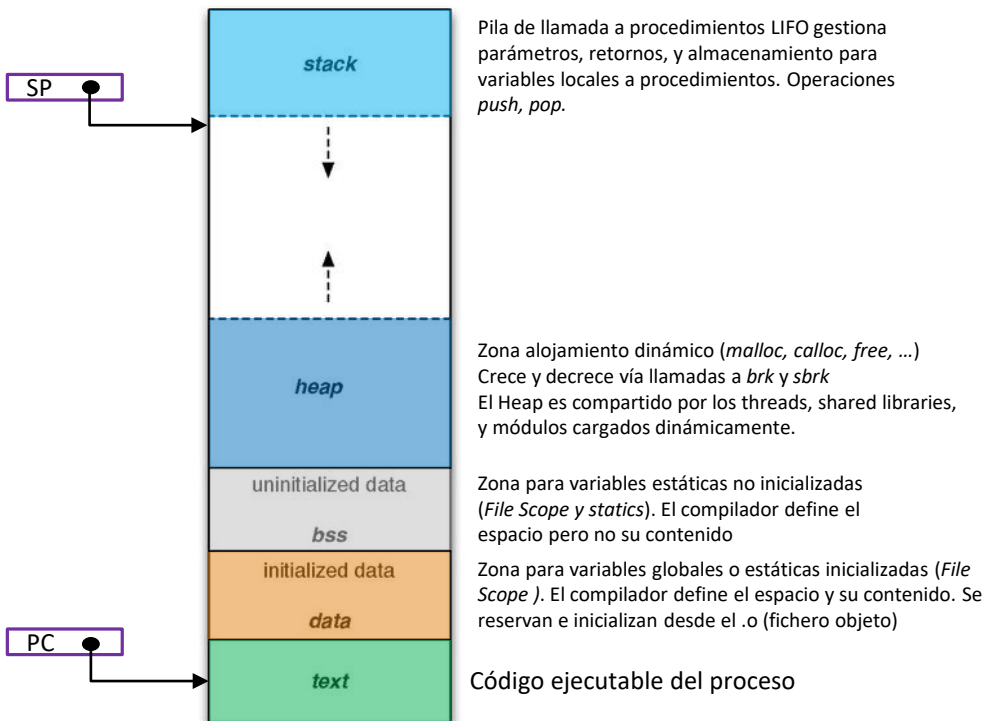
Estructuras de control – Representación física de un proceso

Un proceso debe incluir:

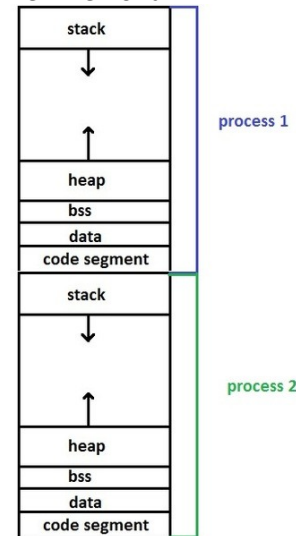
- Un programa o conjunto de programas a ejecutar → Código → Reserva de memoria → Punteros
- Áreas de memoria donde ubicar sus variables y constantes → Datos → Reserva de memoria → Punteros
- Área de memoria donde ubicar la pila de llamadas a procedimientos → Pila → Reserva de memoria → Punteros
- Área de memoria donde ubicar los atributos del proceso → PCB → Reserva de memoria → Punteros

Imagen del proceso

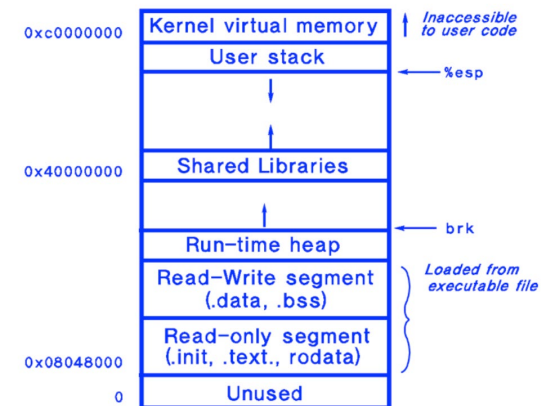
(en memoria)



Varios procesos en memoria>



LINUX Runtime Memory Image



Estructuras de control – Ubicación del proceso

Recordar que un proceso, no necesariamente debe estar completo en memoria (con memoria virtual)

Una vez creado:

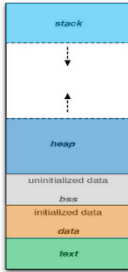
Caso más simple → Completo y contiguo en memoria → Se mantiene en memoria secundaria (Swap)

Para que el S.O. lo pueda gestionar → Hay que cargarlo a memoria principal, al menos una parte

Para que se pueda ejecutar → Debe residir completo en memoria (principal o virtual)

El S.O. debe conocer las ubicaciones en disco, memoria virtual y memoria principal.

S.O. modernos tienen un Hardware de Paginación → **Permite uso de memoria no contigua** → **Ubicación de cada página**
(o segmentación o combinadas)



Tablas de Procesos

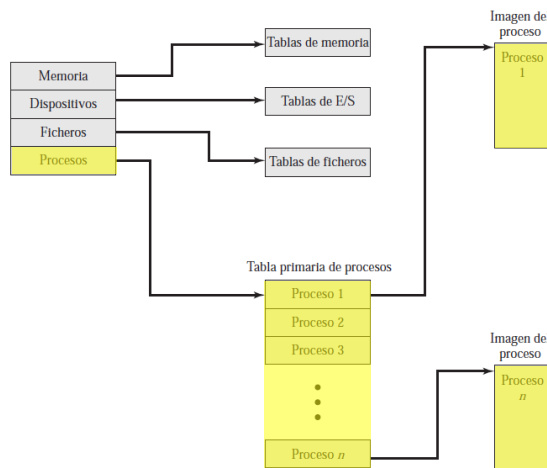


Figura 3.11. Estructura general de las tablas de control del sistema operativo.

Esquema genérico

- ❑ La tabla de procesos contiene una entrada por cada proceso
- ❑ Cada entrada tiene (al menos) un puntero a la imagen del proceso
- ❑ Si la imagen no es contigua, se mantiene una lista enlazada entre ellas o en la entrada

Estructuras de control – Atributos del proceso → PCB

- Toda la meta-información del proceso queda recogida en el PCB
- Cada S.O. organiza el PCB a su modo

Elementos típicos de un Bloque de Control de Procesos (PCB)

Identificación del proceso

Identificadores numéricos

- Identificadores del proceso
- Identificador del padre
- Identificador de usuario

Estado del procesador

Registros visibles por el usuario

Aquellos que son accesibles para el programador (ensamblador) vía el repertorio de instrucciones del microprocesador. Entre 8 y 32 registros. (AX, BX, CX ...)

Registros de estado y control

- Contador de programa
- Registros de condición (signo, zero, acarreo, igual, desbordamiento ...)
- Registros de estado (flags, interrupciones, modo de ejecución...)

Punteros de Pila

- Puntero pila de proceso
- Puntero pila del kernel

Información de Control de Proceso

Información de estado y planificación

- Estado del procesador: Listo, Bloqueado, Suspendido, En Ejecución...
- Prioridad: un o más campos para definir la prioridad (default, actual, greatest...)
- Información de planificación: Depende del algoritmo de planificación (tiempo en ejecución, tiempo en espera, tiempo ultima ejecución, ...)
- Evento: Evento por los que el proceso está bloqueado (fallo de página, E/S, semaphore, pipe, signal, ...)

Estructuras de datos

- Enlace y ubicación en colas y listas.
- Referencias cruzadas entre procesos (comunicación, esperas, padre-hijo, ...)

Comunicación entre procesos

- Flags, mensajes, señales recibidos/enviados
- Procesos involucrados

Privilegios del proceso

- Permisos heredados
- Permisos (modo) de acceso a memoria compartida
- Servicios accesibles ...

Gestión de memoria

- Punteros a tablas de página / segmentos
- Punteros a páginas / segmentos

Recursos

- Ficheros abiertos
- Puertos
- Históricos uso procesador / memoria

Estructuras de control – El rol del PCB

Es la estructura más importante del Sistema Operativo
El conjunto de PCBs definen el estado del sistema



- Cada PCB contiene toda la información que el S.O. necesita
- La información del PCB es leída y **modificada** en cualquier parte del S.O.

Es necesaria la protección del PCB

- Un fallo en una rutina del S.O. puede dañar un PCB y los afectados (referencias cruzadas, listas enlazadas, ...)
- Un cambio en la estructura o semántica del PCB (entre versiones) obliga a rediseñar

Es obligado el acceso al PCB a través de una única rutina o servicio manejador

- Sirve los datos y sincroniza su acceso (mutex)
- Diseñada para un gran rendimiento

Estructuras de control – El PCB enlazado

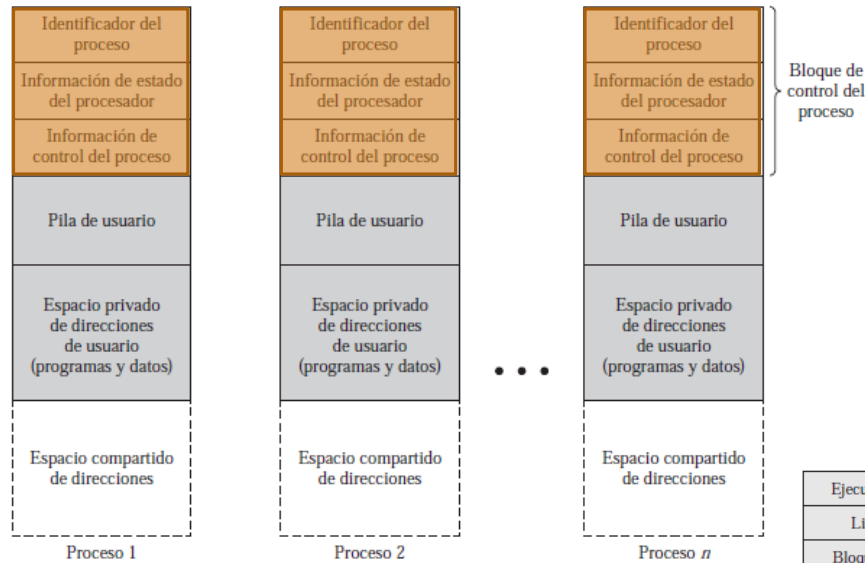
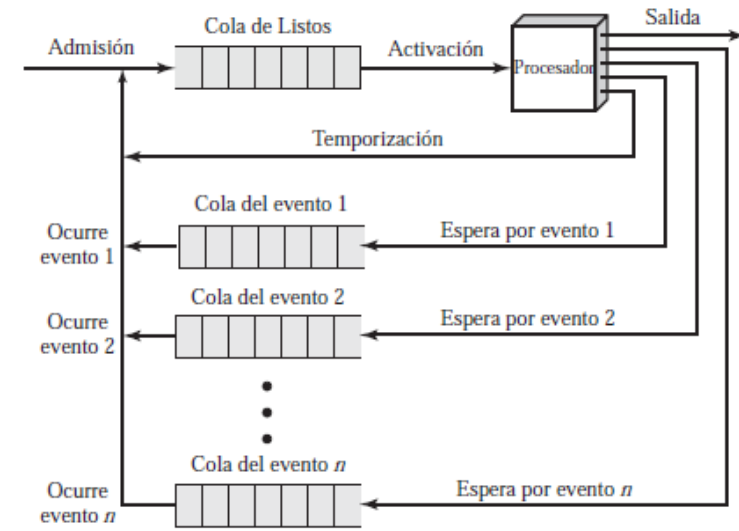


Figura 3.13. Procesos de usuario en memoria virtual.



(b) Múltiples colas de Bloqueados

Figura 3.8. Modelo de colas de la Figura 3.6.

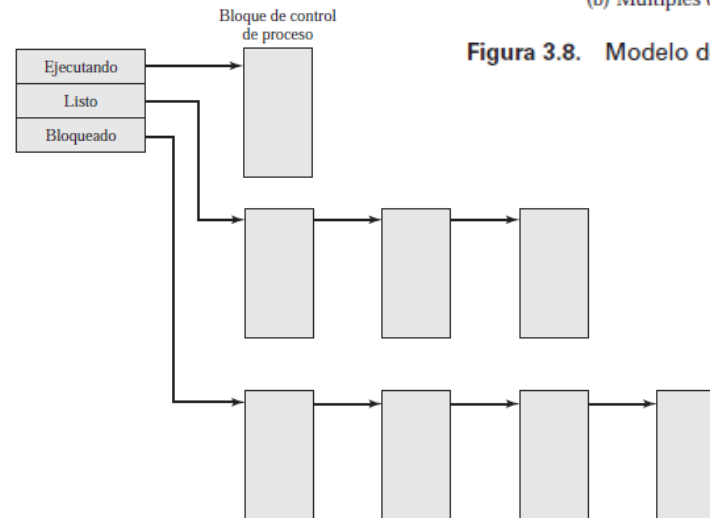


Figura 3.14. Estructuras de listas de procesos.

Estados de los Procesos

El S.O. Planifica la ejecución de los procesos

- Los procesos están en una cola para su de ejecución
→ En la de Listos
- El **Planificador** (Scheduler) selecciona a uno de la cola
→ Varias políticas de planificación (selección)

El Proceso es puesto en ejecución por el S.O.

- El **Activador** (Dispatcher) lo pone en ejecución
→ Utiliza la información del PCB para conseguirlo

El Proceso se ejecuta hasta que:

- Termina → El proceso sale del sistema
- Necesita una operación de E/S
→ Se bloquea hasta que termine la operación
- Necesita un recurso/dato y no está disponible
→ Se bloquea hasta que esté disponible
- Se le acaba el tiempo (quantum)
→ Se pone a la cola de los que están Listos
- Ocurre algo inesperado, una interrupción
→ El S.O. atiende ese algo y le deja continuar
o lo pone en la cola

El S.O. toma el control siempre que pasa algo

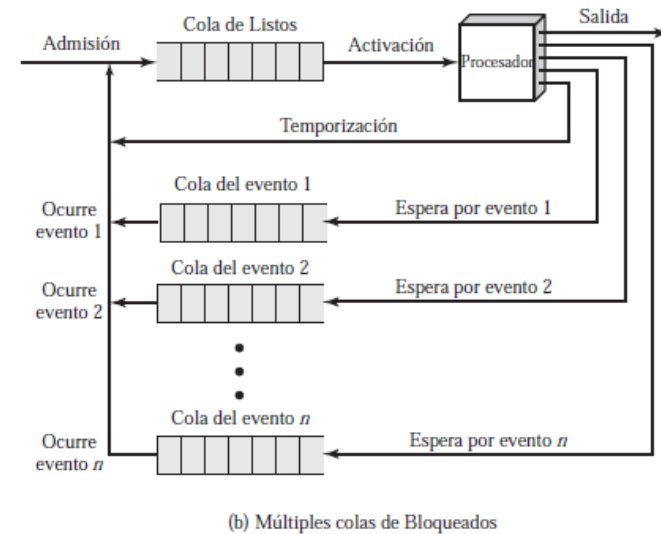
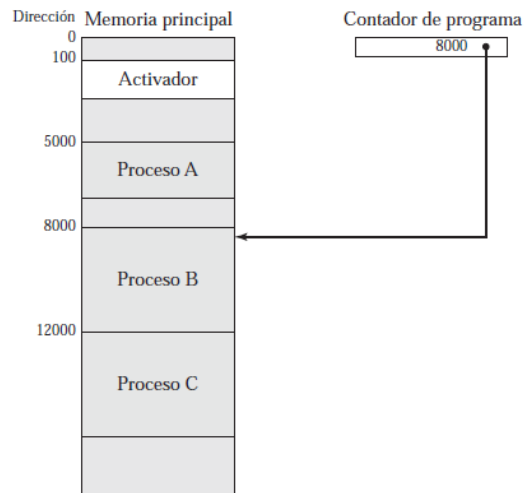


Figura 3.8. Modelo de colas de la Figura 3.6.

Estados de los Procesos

Tres procesos en memoria



Punto de vista de cada proceso

5000	8000	12000
5001	8001	12001
5002	8002	12002
5003	8003	12003
5004		12004
5005		12005
5006		12006
5007		12007
5008		12008
5009		12009
5010		12010
5011		12011
(a) Trazo del Proceso A	(b) Trazo del Proceso B	(c) Trazo del Proceso C

5000 = Dirección de comienzo del programa del Proceso A.
 8000 = Dirección de comienzo del programa del Proceso B.
 12000 = Dirección de comienzo del programa del Proceso C.

Punto de vista del procesador

1	5000		27	12004	
2	5001		28	12005	
3	5002		Planificador		Temporización
4	5003		29	100	
5	5004		30	101	
6	5005		31	102	
Planificador		Quantum = 6 ciclos	32	103	
7	100		33	104	
8	101		34	105	
9	102		35	5006	
10	103		36	5007	
11	104		37	5008	
12	105		38	5009	
13	8000		39	5010	
14	8001		40	5011	
15	8002		Planificador		Temporización
16	8003		41	100	
Planificador		Petición de E/S	42	101	
17	100		43	102	
18	101		44	103	
19	102		45	104	
20	103		46	105	
21	104		47	12006	
22	105		48	12007	
23	12000		49	12008	
24	12001		50	12009	
25	12002		51	12010	
26	12003		52	12011	
			Planificador		Temporización

100 = Dirección de comienzo del programa activador.

Las zonas sombreadas indican la ejecución del proceso de activación;
 la primera y la tercera columna cuentan ciclos de instrucciones;
 la segunda y la cuarta columna las direcciones de las instrucciones que se ejecutan

Estados de los Procesos – Creación / Terminación

Se va a añadir un nuevo proceso

- El S.O. crea y actualiza todas las estructuras de datos (PCB, inserción en tabla de procesos,...)
- Crea la Imagen del proceso (cargando sus código, datos, etc...)
- Añade el proceso en la cola correspondiente (Listos, nuevos...)

Eventos que provocan la creación de un proceso

- **Emisión de un nuevo proceso por lotes (batch)**
Existe un sistema de carga de trabajos (Job Queues – Colas de trabajos) y decide que puede comenzar con un nuevo trabajo
- **Sesión Interactiva**
Un usuario desde terminal o GUI quiere ejecutar un programa
- **Para ofrecer un servicio**
Crea un servicio que el proceso en ejecución o el sistema demanda (transparente dicho proceso)
- **Por solicitud del proceso en ejecución**
El proceso que está en ejecución demanda al S.O. la creación de un proceso (un proceso hijo).

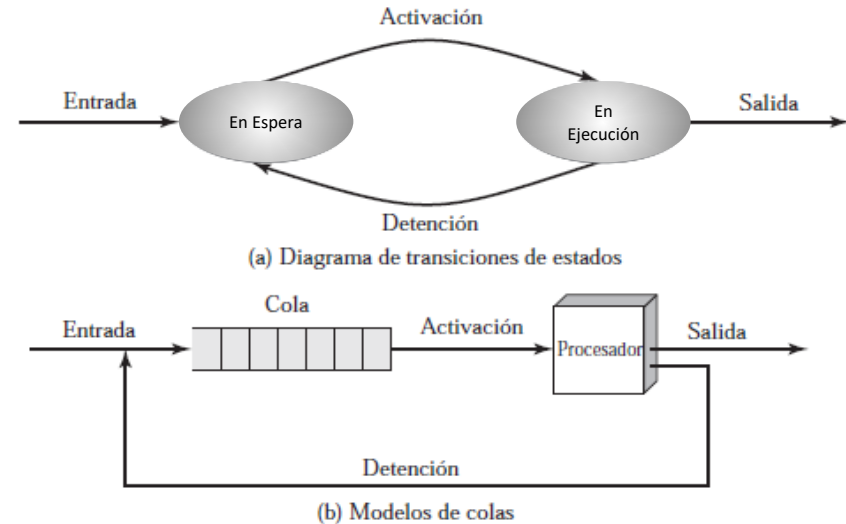


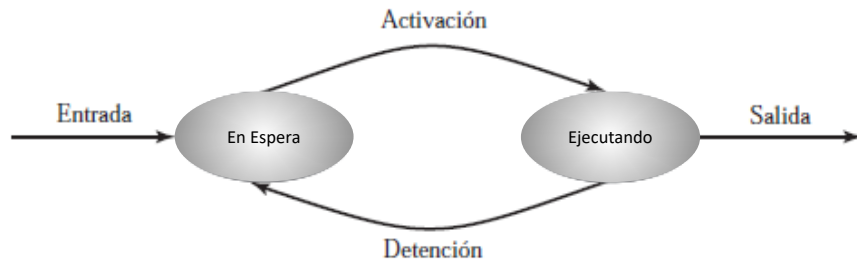
Figura 3.5. Modelo de proceso de dos estados.

Estados de los Procesos – Creación de Procesos

Eventos que provocan la terminación de un proceso

- **Finalización normal** : El proceso le indica al S.O. que ha terminado `exit()`;
- **Límite de tiempo excedido** : Si se han establecido límites y el proceso supera el tiempo permitido, tiempo de ejecución, tiempo desde la última interacción del usuario, ...
- **Memoria no disponible** : Solicita más memoria que la disponible, o se traspasan los tamaños de pila y/o heap.
- **Violaciones de frontera** : Trata de acceder a una zona de memoria fuera de sus límites
- **Error de protección** : Trata de acceder a un recurso para el que no tiene permisos. O accediendo en forma inapropiada, por ejemplo escribir en un fichero de sólo lectura.
- **Error aritmético** : Operación de cálculo no permitida, por ejemplo división por cero
- **Límite de tiempo** : Ha esperado por un evento más tiempo del fijado
- **Fallo de E/S** : Error en una operación de E/S. Ej: Fichero no encontrado, fallo tras n intentos de escritura, dispositivo fuera de línea,
- **Instrucción no válida** : Intento de ejecutar una instrucción inexistente, salto a datos e intento de ejecución.
- **Instrucción privilegiada** : Intento de ejecutar, en modo usuario, una instrucción reservada al kernel.
- **Uno inapropiado de datos** : Acceso a datos no inicializados, mal referenciados, sin reserva de espacio,
- **Intervención del operador (root)** : El usuario root decide matar un proceso, posible interbloqueo...
- **Terminación del padre** : Cuando el proceso padre termina, el S.O. mata a todos sus hijos. Cierre de un terminal, cierre de sesión RDP, simple finalización del padre, ...
- **Solicitud del proceso padre** : Por decisión del proceso padre, tiene privilegios para matar a sus hijos.

Estados de los Procesos – Modelo de 5 estados



(a) Diagrama de transiciones de estados

Este modelo no es apropiado

Los procesos *En Espera* lo están por diversos motivos

El planificador debe recorrer la cola para seleccionar los que están preparados

Entra en el sistema. Se crean e inicializan sus estructuras. A la espera de recursos necesarios o por limitación de recursos en el sistema

Tiene todos los recursos asignados y está listo para ejecutar, sólo le falta la CPU

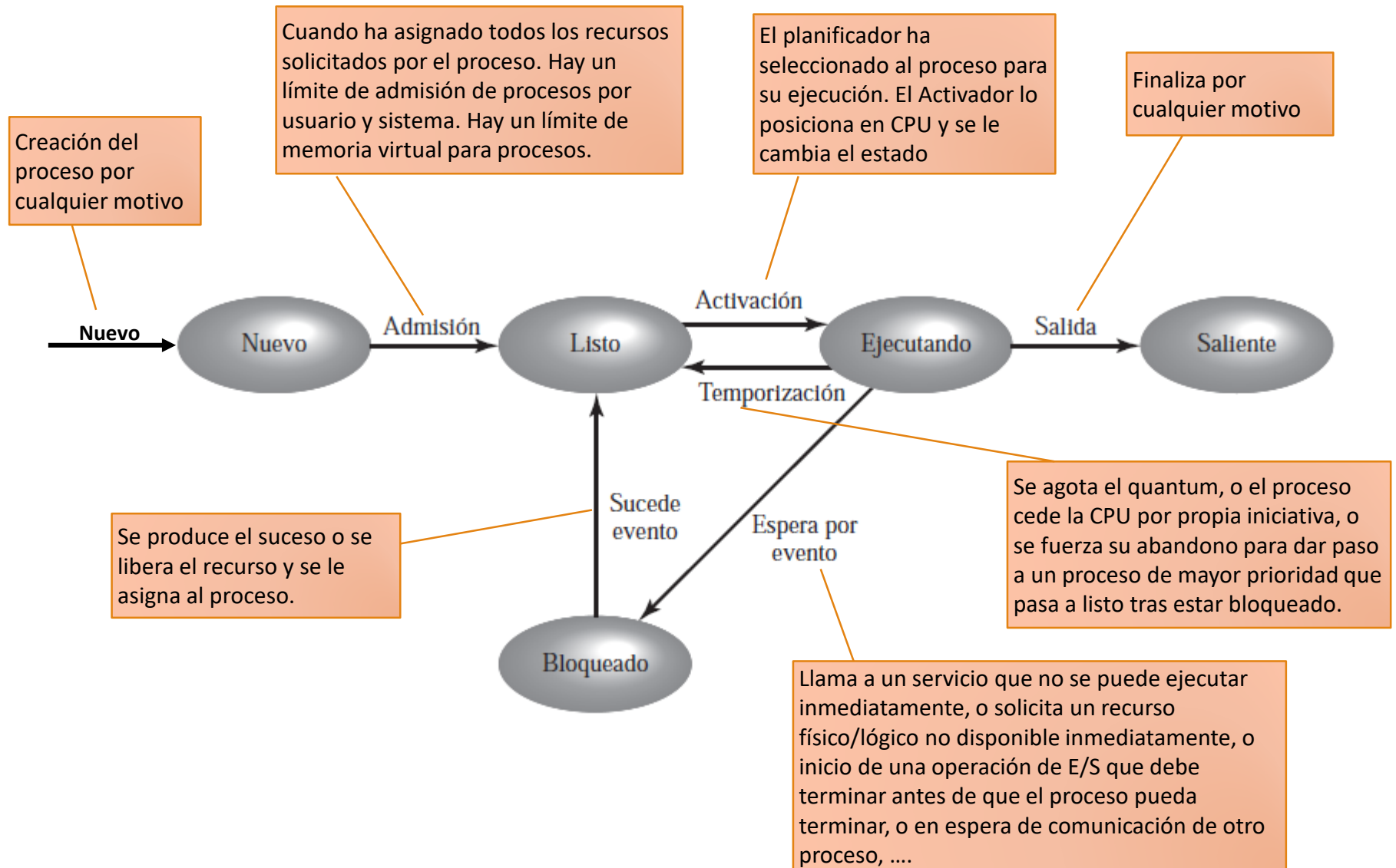
El proceso tiene la CPU, está en ejecución. Suponemos un único procesador.

Ya no podrá ejecutar. Su información está disponible. Motivo finalización, contabilidad, auditoría por otros procesos, etc..

No puede ejecutar hasta que ocurra el evento que lo bloqueó.

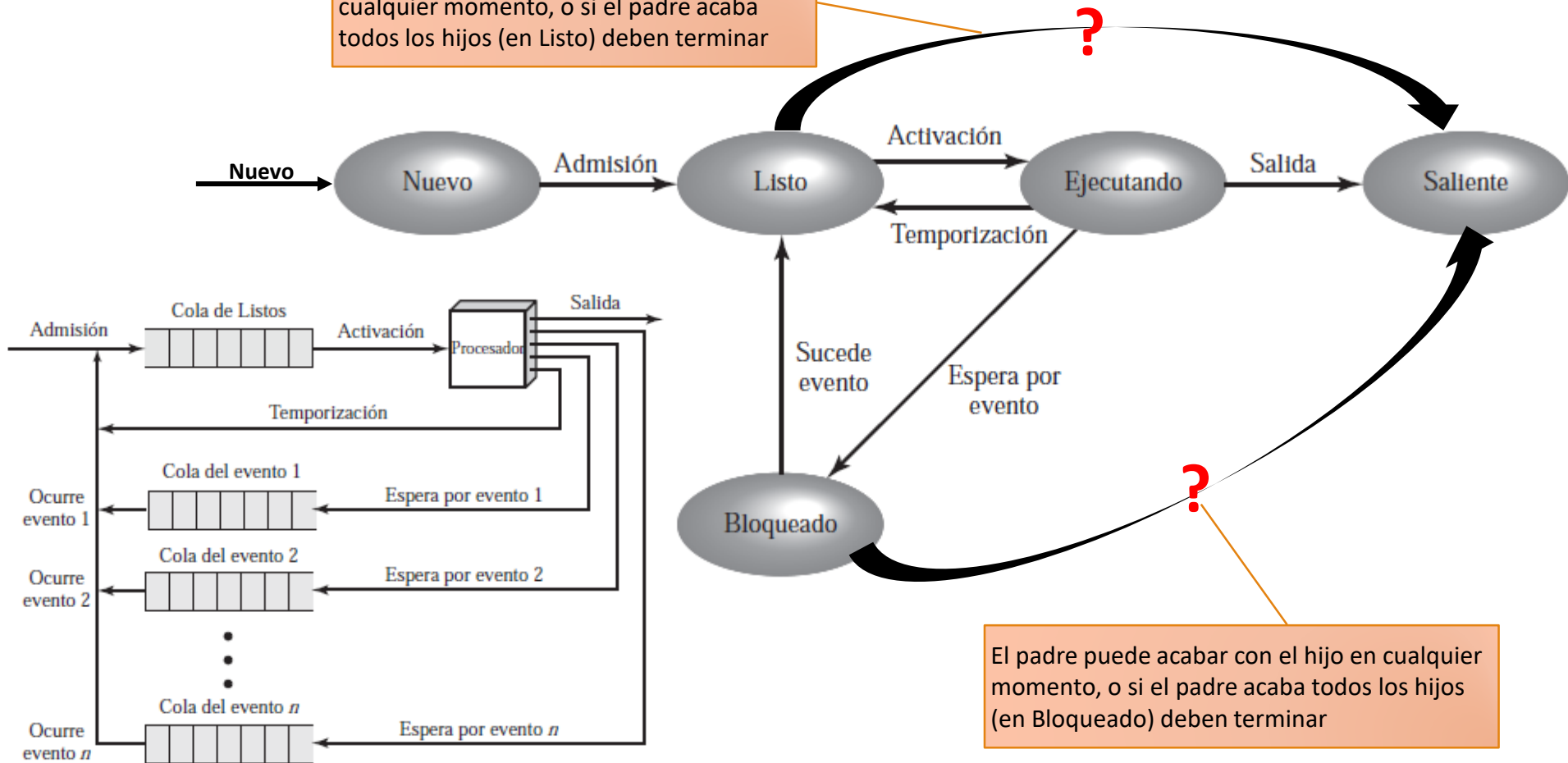


Estados de los Procesos – Modelo de 5 estados



Estados de los Procesos – Modelo de 5 estados

El padre puede acabar con el hijo en cualquier momento, o si el padre acaba todos los hijos (en Listo) deben terminar



(b) Múltiples colas de Bloqueados

Figura 3.8. Modelo de colas de la Figura 3.6.

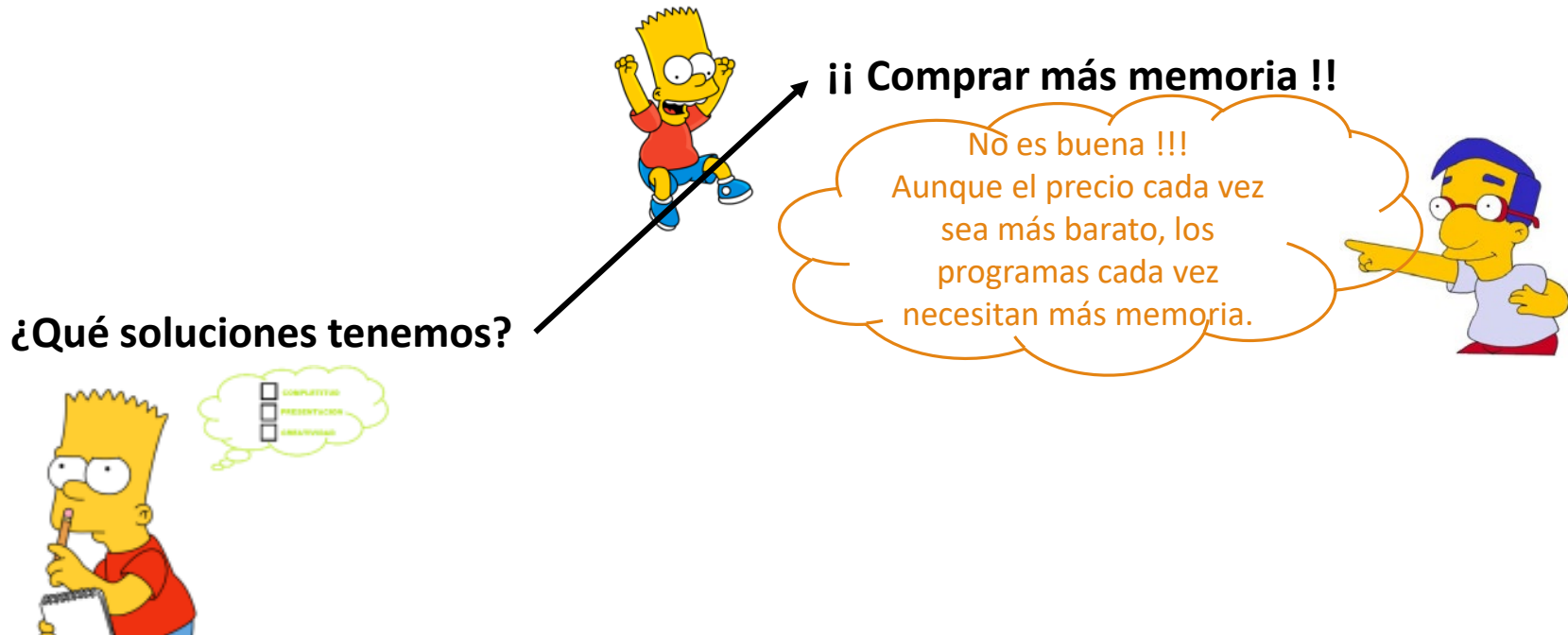
Si tengo procesos con distintas prioridades en Listo, ¿cómo mejoraría la búsqueda?

Estados de los Procesos – Intercambio o Swapping

Supongamos que no existe memoria virtual → El proceso debe estar completo en memoria
Las operaciones de E/S son muchísimo más lentas que el procesador



**El sistema evolucionará a que la mayoría de sus procesos estarán bloqueados.
(Sistema Multiprogramado → Ocioso !!!)**



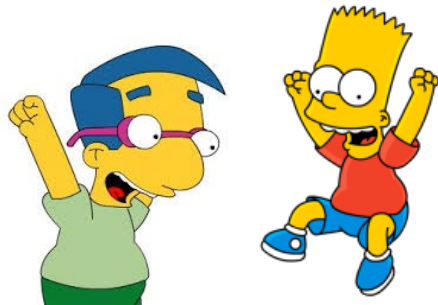
Estados de los Procesos – Intercambio o Swapping

Supongamos que no existe memoria virtual → El proceso debe estar completo en memoria
Las operaciones de E/S son muchísimo más lentas que el procesador



**El sistema evolucionará a que la mayoría de sus procesos estarán bloqueados.
(Sistema Multiprogramado → Ocioso !!!)**

¿Qué soluciones tenemos?



ii Comprar más memoria !!

Inventamos el Intercambio

Intercambio

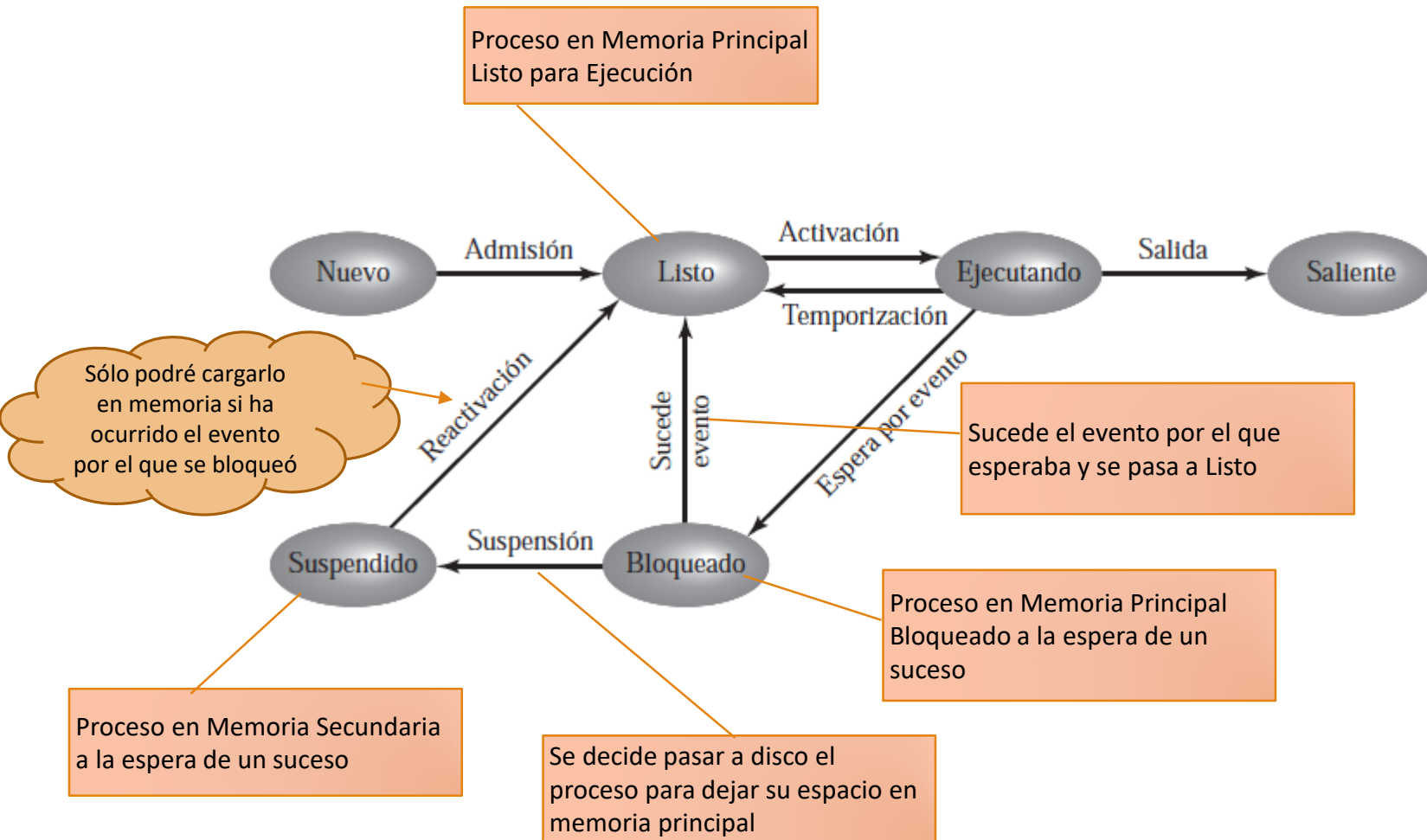
- ☐ Un proceso está bloqueado
 - Lo llevo a Disco (Swap File)
 - Lo suspendo
- ☐ Queda sitio para otro, Nuevo o Suspendido
- ☐ El intercambio es una op. de E/S
 - A pesar de eso es más eficiente
 - Sobre todo si usamos el disco más rápido

Esta Si !!!!



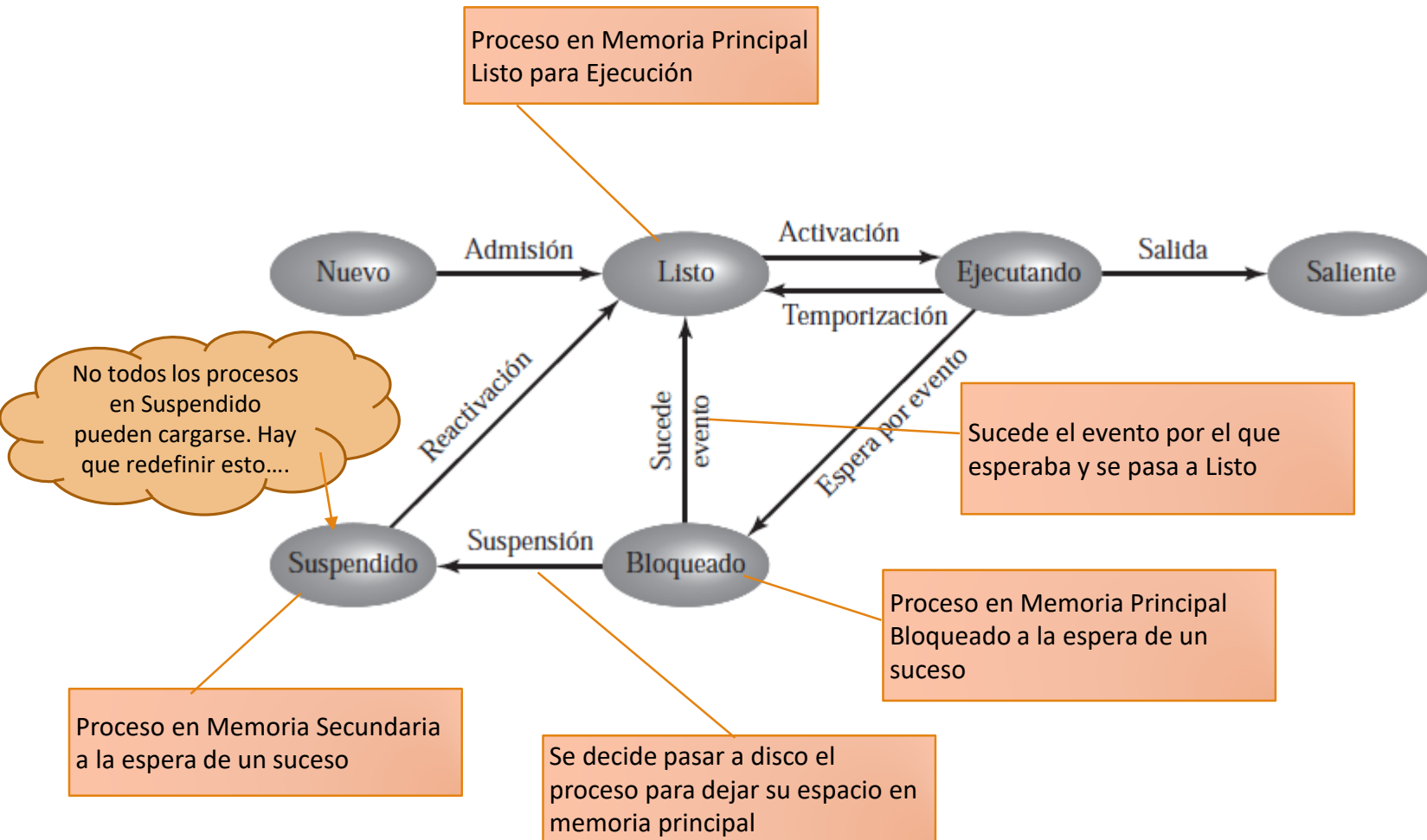
Estados de los Procesos – Intercambio o Swapping

Aparece un nuevo estado → Suspendido



Estados de los Procesos – Intercambio o Swapping

Aparece un nuevo estado → Suspendido



Estados de los Procesos – Intercambio o Swapping

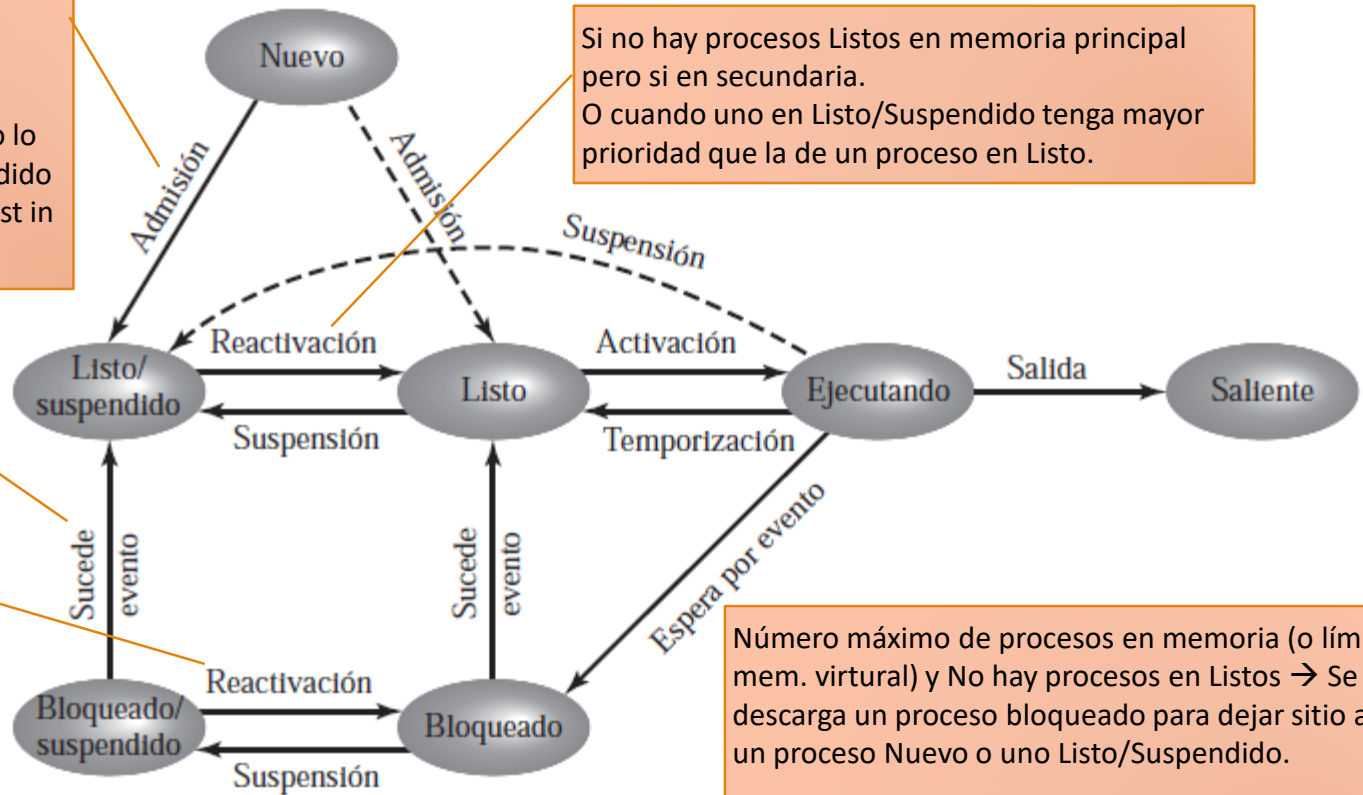
Una cosa es la Suspensión y otro el Bloqueo

→ 4 combinaciones → (Bloqueado, Bloqueado/Suspendido, Listo, Listo/Suspendido)

Para mantener un pool de procesos disponibles para ejecución pero sin sobrecargar la memoria principal cuando esta es escasa. Esto aumenta la sobrecarga cuando lo habitual sea pasar de Listo/Suspendido a Listo en vez de crear procesos "just in time" cuando no hayan Listos.

Sucede el evento por el que esperaba y se pasa a Listo/Suspendido

El proceso suspendido tiene mayor prioridad que ninguno de los bloqueados en el momento que se libera memoria por un proceso terminado. Se esta suponiendo que pronto ocurrirá el suceso o se liberará el recurso que lo bloqueó.



Si no hay procesos Listos en memoria principal pero si en secundaria. O cuando uno en Listo/Suspendido tenga mayor prioridad que la de un proceso en Listo.

Número máximo de procesos en memoria (o límite mem. virtual) y No hay procesos en Listos → Se descarga un proceso bloqueado para dejar sitio a un proceso Nuevo o uno Listo/Suspendido.

Aun cuando haya procesos Listos se puede dar la transición porque el proceso en ejecución demanda memoria, siendo necesario liberar memoria.

Estados de los Procesos – Intercambio o Swapping

Una cosa es la Suspensión y otro el Bloqueo

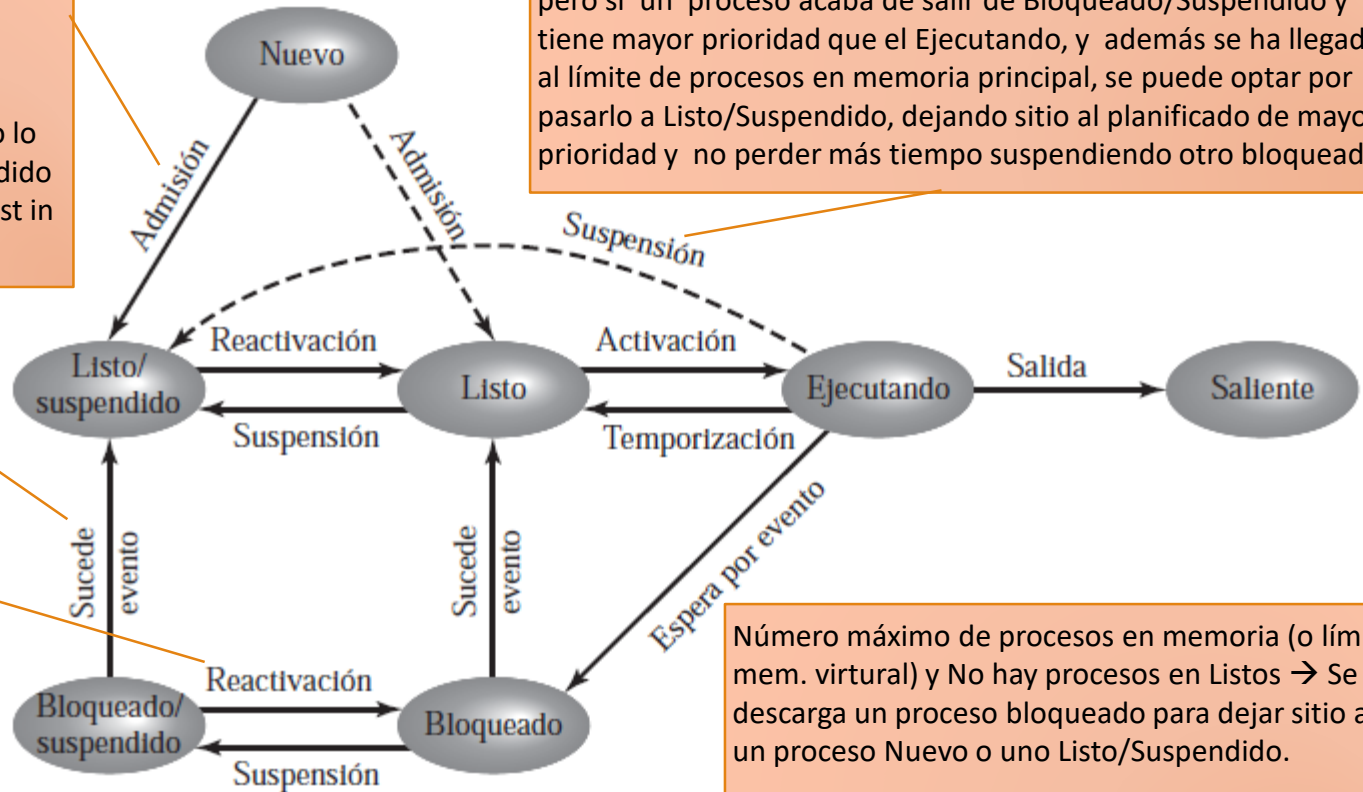
→ 4 combinaciones → (Bloqueado, Bloqueado/Suspendido, Listo, Listo/Suspendido)

Para mantener un pool de procesos disponibles para ejecución pero sin sobrecargar la memoria principal cuando esta es escasa. Esto aumenta la sobrecarga cuando lo habitual sea pasar de Listo/Suspendido a Listo en vez de crear procesos "just in time" cuando no hayan Listos.

Sucede el evento por el que esperaba y se pasa a Listo/Suspendido

El proceso suspendido tiene mayor prioridad que ninguno de los bloqueados en el momento que se libera memoria por un proceso terminado. Se esta suponiendo que pronto ocurrirá el suceso o se liberará el recurso que lo bloqueó.

Normalmente a Listo (quantum o planificación expulsiva prioridad) pero si un proceso acaba de salir de Bloqueado/Suspendido y tiene mayor prioridad que el Ejecutando, y además se ha llegado al límite de procesos en memoria principal, se puede optar por pasarlo a Listo/Suspendido, dejando sitio al planificado de mayor prioridad y no perder más tiempo suspendiendo otro bloqueado.



Número máximo de procesos en memoria (o límite mem. virtual) y No hay procesos en Listos → Se descarga un proceso bloqueado para dejar sitio a un proceso Nuevo o uno Listo/Suspendido.

Aun cuando haya procesos Listos se puede dar la transición porque el proceso en ejecución demanda memoria, siendo necesario liberar memoria.

Estados de los Procesos – Intercambio o Swapping

Supongamos que no existe memoria virtual → El proceso debe estar completo en memoria

Las operaciones de E/S son muchísimo más lentas que el procesador

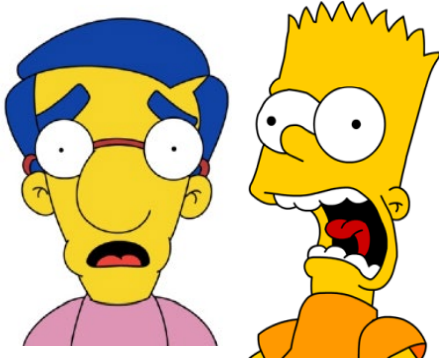


El sistema evolucionará a que la mayoría de sus procesos estarán bloqueados.
(Sistema Multiprogramado → Ocioso !!!)

ii Comprar más memoria !!

Inventamos el Intercambio

¿Qué soluciones tenemos?



Cuando Bart y Milhouse inventaron el Intercambio, estaban asumiendo

ii que no existía Memoria Virtual !!

Si tenemos Memoria Virtual

(no necesariamente todo el proceso en memoria → Paginación y/o Segmentación)

¿Tiene sentido el Intercambio y por tanto la Suspensión?

Con la paginación hay más sitio libre en memoria principal, ¿no?

Estados de los Procesos – Intercambio o Swapping

Supongamos que no existe memoria virtual → El proceso debe estar completo en memoria
Las operaciones de E/S son muchísimo más lentas que el procesador



El sistema evolucionará a que la mayoría de sus procesos estarán bloqueados.
(Sistema Multiprogramado → Ocioso !!!)

ii Comprar más memoria !!

Inventamos el Intercambio

¿Qué soluciones tenemos?



¿Tiene sentido el Intercambio y por tanto la Suspensión?
on la paginación hay más sitio libre en memoria principal, ¿no?

iii Pues si !!!

Con Memoria Virtual y con demasiados procesos parcialmente en memoria (todos con trozos muy pequeños), llega un momento en el que cae el rendimiento drásticamente, muchos fallos de página
→ El S.O. debe recurrir al Swapping Explícito para mantener el rendimiento

Estados de los Procesos – Usos de la Suspensión

Un proceso suspendido no está en Memoria Principal

Un proceso Suspendido Cumple:

1. No está disponible inmediatamente para su ejecución
2. Puede estar o no a la espera de un evento de desbloqueo
→ Independencia entre Bloqueo y Suspensión
3. El proceso fue suspendido por un agente:
 - Él mismo
 - El proceso padre
 - El Sistema Operativo
4. El proceso no puede ser recuperado de la Suspensión hasta que el agente lo indique.

Estados de los Procesos – Motivos para Suspensión

Swapping o Intercambio

El sistema operativo necesita liberar suficiente memoria principal para cargar un proceso que está listo para ejecutarse.

Otra razón del SO

El sistema operativo puede suspender a un proceso subordinado o de utilidad, o a un proceso que se sospecha que sea el causante de un problema.

Solicitud de un usuario interactivo

Un usuario puede querer suspender la ejecución de un programa con fines de depuración o en conexión con el uso de un recurso.

Temporización

Un proceso puede ejecutarse periódicamente (por ejemplo, un proceso de contabilidad o de supervisión del sistema) y puede ser suspendido mientras espera el siguiente intervalo de tiempo.

Solicitud del proceso padre

Un proceso padre puede querer suspender la ejecución de un descendiente para examinar o modificar el proceso suspendido o para coordinar la actividad de varios descendientes.

Fin

UNIDAD 3 – SISTEMA DE PROCEOSS

PROCESOS, ESTRUCTURAS Y ESTADOS