

ADMINISTRACIÓN DE PROCESOS Y DEL PROCESADOR

Competencia:

Comprende las técnicas de administración de procesos para crear procesos empleando los mecanismos que presenta el sistema operativo para la comunicación y sincronización

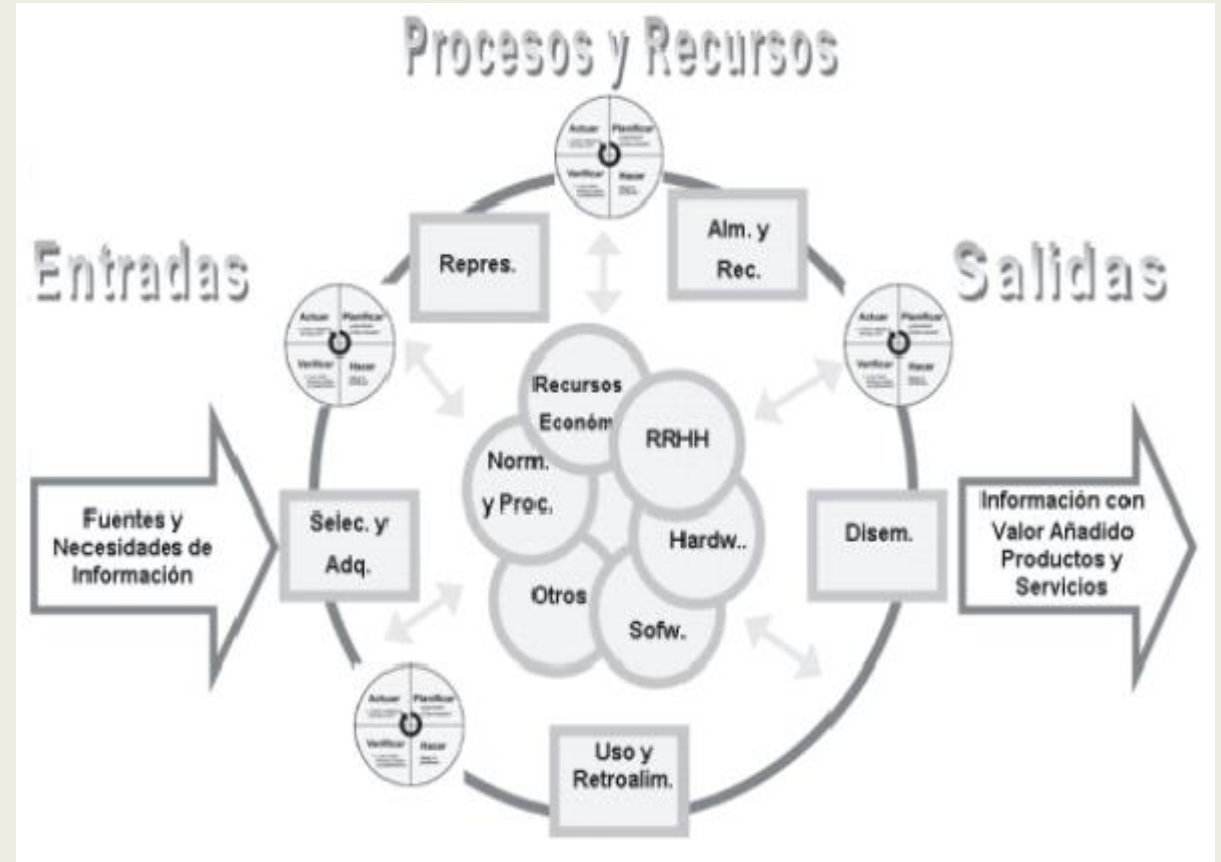


Temas

- 2.1. Concepto de proceso
- 2.2. Estados y transiciones de los procesos
- 2.3. Procesos ligeros: Hilos o hebras
- 2.4. Concurrencia y secuenciabilidad
- 2.5. Niveles, objetivos y criterios de planificación
- 2.6. Técnicas de administración del planificador

Concepto de Proceso

- Es una abstracción de un programa en ejecución y es la unidad de trabajo del sistema
- Es un programa en ejecución.
- Un proceso no es más que un programa en ejecución, e incluye los valores actuales del contador del programa, los registros y las variables.



El SO tiene las siguientes funciones

- Crear y eliminar los procesos del usuario y del sistema
- Suspende y reanuda la ejecución de los procesos
- Proporcionar mecanismo para la sincronización y planificación de procesos
- Preparación de tareas y programas: Se transfieren los programas ejecutables de usuario desde el almacenamiento secundario a la memoria principal
- Planificación del procesador: Se controla la utilización del procesador por parte de los diferentes procesos que se encuentren en el sistema en cada momento
- Asignación de periféricos: Se transforman las solicitudes simbólicas de periféricos en asignaciones físicas
- Relanzamiento de programas: Si durante la ejecución de un programa se produce alguna interrupción, el SO establece un punto de control, a partir del cual se pueda volver al estado inicial.

Un proceso es un concepto manejado por el SO que consiste en el conjunto formado por:

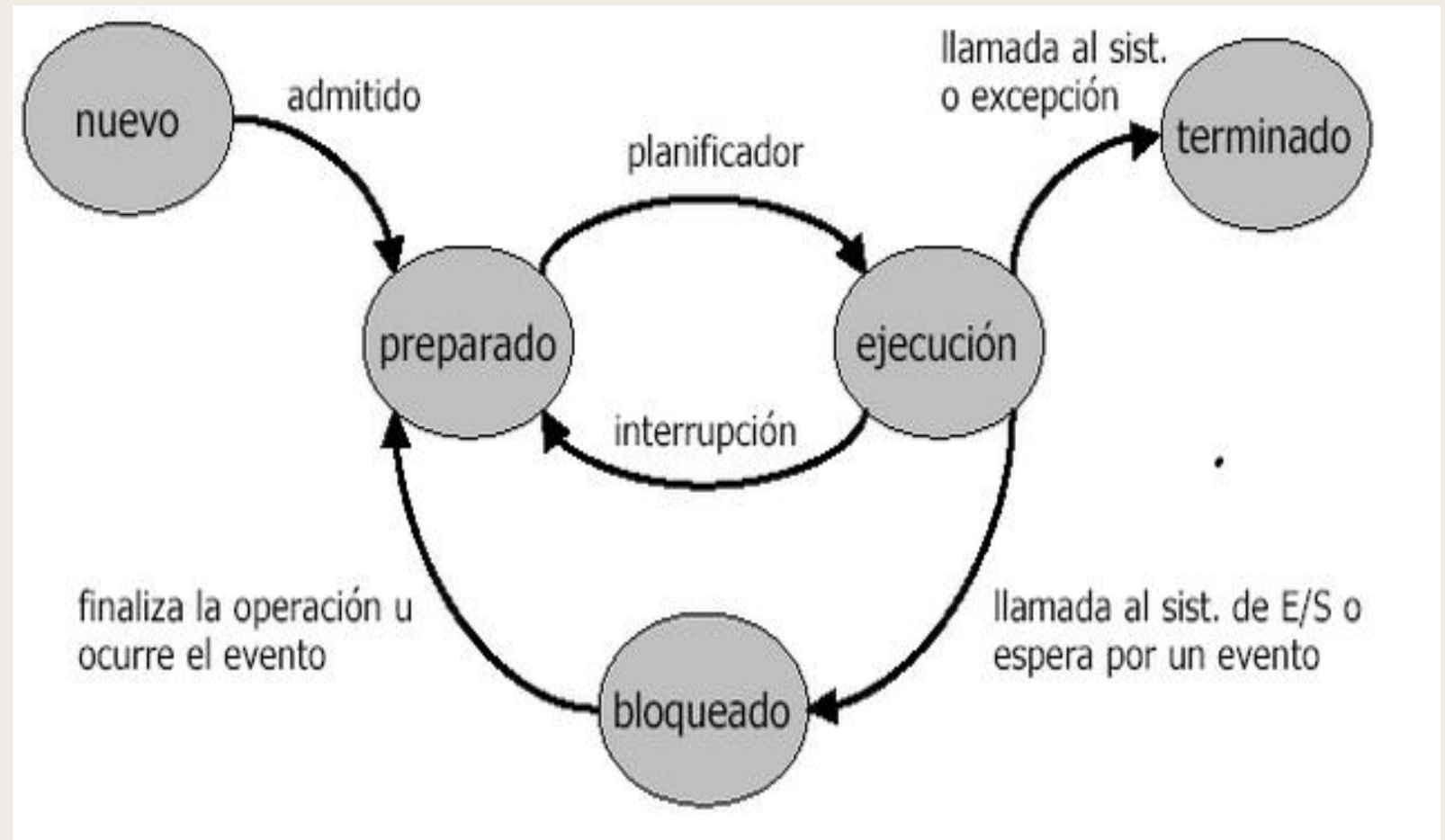
- Las instrucciones de un programa destinadas a ser ejecutadas por el microprocesador
- Su estado de ejecución en un momento dado, esto es, los valores de los registros de la CPU para dicho programa
- Su memoria de trabajo, es decir, la memoria que ha reservado y sus contenidos
- Otra función que permite al sistema operativo su planificación.

Un proceso Incluye:

- Program counter (Contador de programa)
- Stack (pila)
- Data section (segmento de datos)

Estados de un proceso

- Nuevo
- En ejecución
- En espera
- Preparado
- Terminado



Un proceso puede ser finalizado por:

- Terminación normal
- Terminación por error
- Error fatal
- Terminación por otro proceso



. Estados y transiciones de los procesos

- Un proceso pasa de un estado a otro desde su creación hasta su destrucción.



PCB (Bloque de control de procesos)

Procesos Ligeros: Hilos o hebras

- Un proceso puede contener un solo flujo de ejecución, como ocurre en los procesos clásicos, o mas de un flujo de ejecución(procesos ligeros).
- Hola mi amor att Diego

Dentro de un proceso puede haber uno o más hilos de control cada uno con:

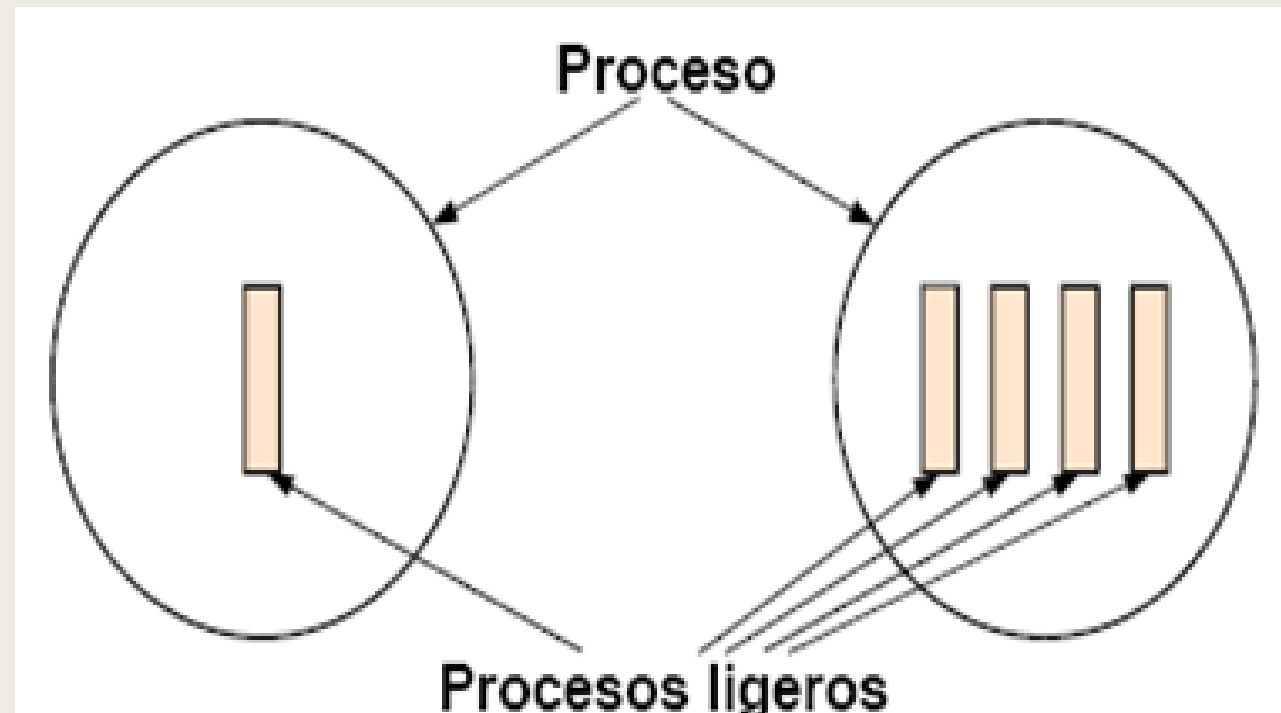
- Un estado de ejecución (en ejecución, listo, bloqueado).
- Un contexto de procesador, que se salva cuando no esté ejecutándose.
- Una pila de ejecución.
- Algún almacenamiento estático para variables locales.
- Acceso a la memoria y a los recursos de ese trabajo que comparte con los otros hilos.
- siuuuuuuuuuu

Diferencia entre proceso e Hilo

- Un proceso es una entidad relativamente independiente que dispone de su propio espacio de direcciones, su propia información de estado y que utiliza los mecanismos de comunicación entre procesos que le proporciona el sistema operativo para comunicarse con otros procesos.
- Por otro lado, un hilo es una entidad más reducida capaz de convivir junto a otros hilos bajo el contexto de un único proceso, permitiendo compartir la información de estado, el área de memoria y/o los recursos asociados a ese proceso.

Procesos ligeros

- Un proceso ligero (thread o hebra) es un programa en ejecución que comparte la imagen de la memoria y otras informaciones con otros procesos ligeros.

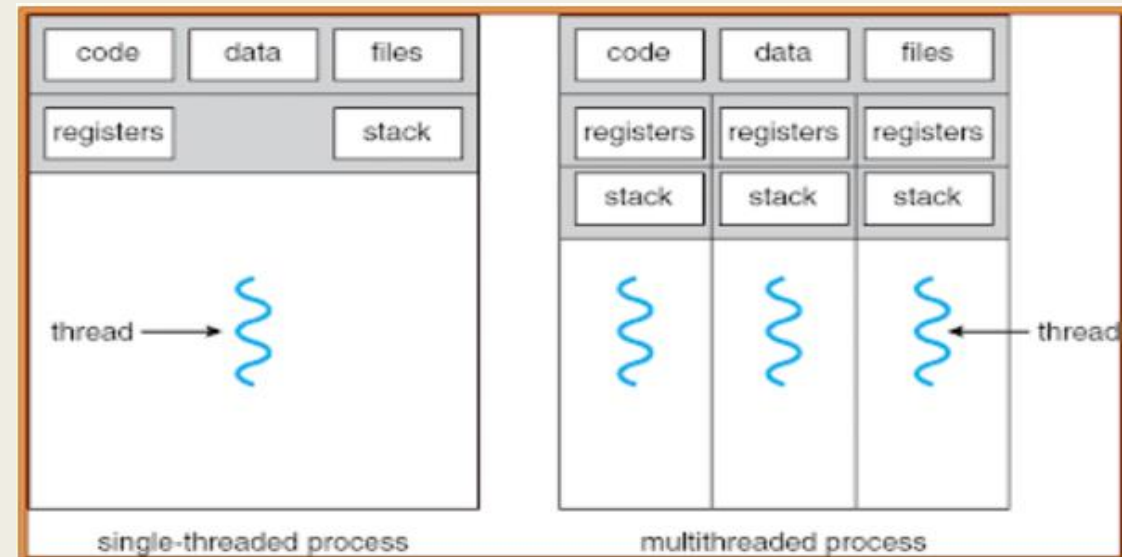


Procesos ligeros

- Los procesos ligeros son una unidad básica de utilización de la CPU consistente en un juego de registros y un espacio de pila. Comparte el código, los datos y los recursos con sus hebras pares

Una tarea (o proceso pesado) está formada ahora por una o más hebras

Una hebra sólo puede pertenecer a una tarea



Características

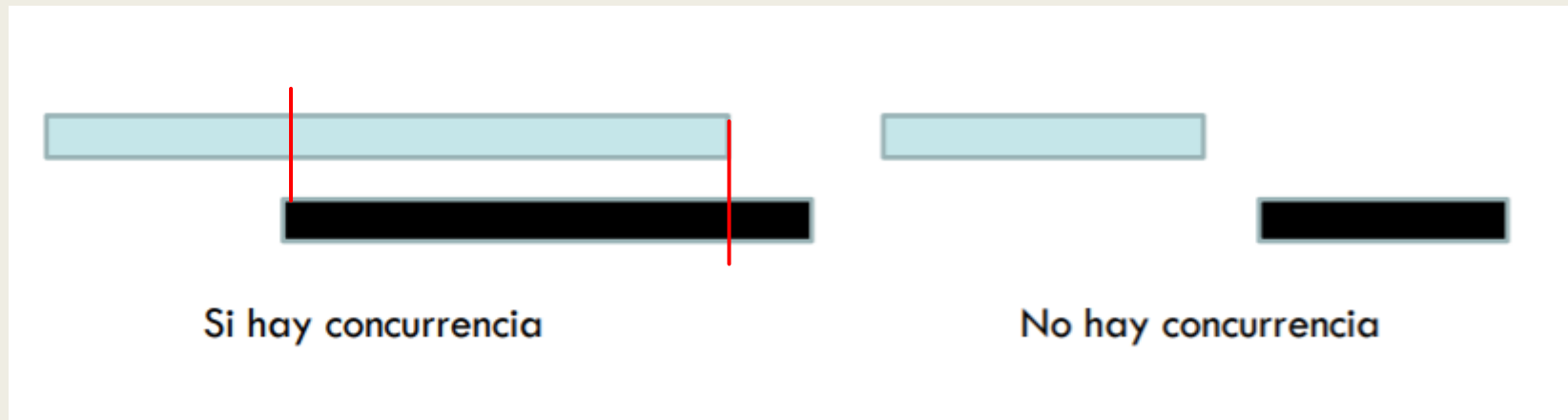
- Se comparten recursos. La compartición de la memoria permite a las hebras pares comunicarse sin usar ningún mecanismo de comunicación inter-proceso del SO.
- La conmutación de contexto es más rápida gracias al extenso compartir de recursos
- No hay protección entre las hebras. Una hebra puede escribir en la pila de otra hebra del mismo proceso

Estado de los procesos ligeros



Concurrencia y Secuenciabilidad

- Concurrencia: Es la existencia de varias actividades ejecutándose simultáneamente, y necesitan sincronizarse para actuar conjuntamente.
- Dos procesos son concurrentes cuando se ejecutan de manera que sus intervalos de ejecución se solapan.



Ventajas de la ejecución concurrente

- Facilita la programación:
 - *Diversas tareas se pueden estructurar en procesos separados*
- Acelera la ejecución de cálculos
 - *División de cálculos en procesos ejecutados en paralelo*
- Mejora la interactividad de las aplicaciones
 - *Se pueden separar las tareas de procesamiento de las tareas de atención a usuarios*
- Mejora el aprovechamiento de la CPU
 - *Se aprovechan las fases de E/S de una aplicación para procesamiento de otras*

Regiones Críticas

- Como evitar la condición de carrera?....
- Es necesaria la Exclusión Mutua
- Es parte del programa en la que se accede a la memoria compartida se conoce como región crítica o sección crítica.

Condiciones para tener una buena solución para el acceso a la sección crítica

- No puede haber dos procesos de manera simultánea dentro de sus regiones críticas.
- No pueden hacerse suposiciones acerca de las velocidades o el número de CPUs.
- Ningún proceso que se ejecute fuera de su región crítica puede bloquear otros procesos.
- Ningún proceso tiene que esperar para siempre para entrar a su región crítica

Tarea

1. Alternancia
2. Variables de Cerradura
3. TSL
4. Solución de Petterson
5. Semaforos
6. Monitores
7. Contadores
8. Cena de los filósofos
9. Problema del barbero dormilón
10. Problema de Lectores y escritores

Sincronización

Cuando los procesos se reúnen para realizar tareas en conjunto, a este tipo de relación se le llama procesos cooperativos.

Para lograr la comunicación, los procesos deben sincronizarse

Sincronización

Es la transmisión y recepción de señales que tienen por objeto llevar a cabo el trabajo de un grupo de procesos cooperativos.

Es la coordinación y cooperación de un conjunto de procesos para asegurar la comparación de recursos de cómputo.

Sincronización

- Es necesaria para prevenir y/o corregir errores de sincronización debido al acceso concurrente a recursos compartidos como:
 - *Estructuras de datos*
 - *Dispositivos de E/S de procesos*
- Permite intercambiar señales de tiempo (arranque/parada) entre procesos cooperantes para garantizar las relaciones específicas de precedencia impuestas por el problema que se resuelve.
- Alan joto

Sincronización

Sin una sincronización adecuada entre procesos, la actualización de variables compartidas puede inducir a errores de tiempo relacionados con la concurrencia que son con frecuencia difíciles de depurar.

Sincronización

Una de las causas principales de este problema es realizar actualizaciones de variables compartidas de manera mutuamente exclusiva. Se puede mejorar permitiendo que a lo más un proceso entre a la vez en la sección crítica en la que se actualiza una variable compartida o estructura de datos en particular.

Relación entre procesos concurrentes

En la compartición de recursos de un solo sistema, los procesos concurrentes compiten unos con otros por la asignación de los recursos del sistema necesarios para sus operaciones.

La cooperación es normal entre los procesos creados como resultado de una división explícita de tareas

Relación entre procesos concurrentes

- Los procesos cooperativos intercambian datos y señales de sincronización necesarias para organizar su progreso colectivo.
- La competición requiere una cuidadosa asignación y protección de los recursos
- La cooperación depende de la existencia de un mecanismo para controlar los datos compartidos y el intercambio de señales de sincronización.

Procesos Cooperativos

Cualquier proceso que comparte datos con otros procesos es un proceso cooperativo.

Razones para proporcionar un ambiente que permita la cooperación entre procesos

- Compartir información:
 - *Archivos compartidos*
- Aceleración de cálculos:
 - *Descomponer el tareas, cada una ejecutarse en paralelo*
- Modularidad:
 - *Dividir en módulos las funciones del sistema en procesos distintos o hilos*
- Conveniencia:
 - *Editar*
 - *Imprimir*
 - *Etc. (al mismo tiempo)*

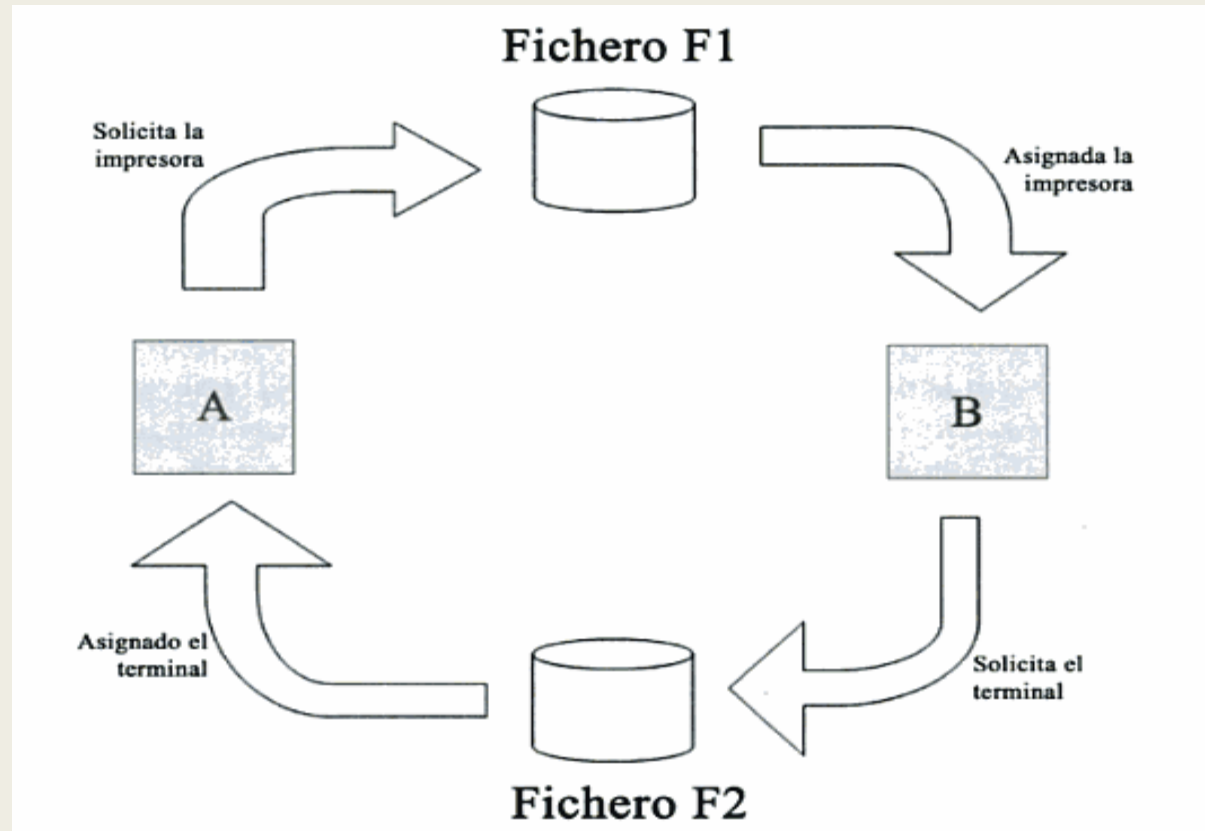
Problemas clásicos de comunicación entre procesos

El problema de la cena de los filósofos

El problema de lectores y escritores

El problema del barbero dormilón

Interbloqueo



Interbloqueo

También conocido como abrazo mortal

Se presenta cuando dado un conjunto determinado de procesos, cada uno de ellos espera que tenga lugar un suceso, que solo puede ser producido por alguno de los procesos de dicho conjunto.

Condiciones del interbloqueo

No Apropiación: Los procesos no podrán adueñarse de los recursos que ya están siendo utilizados por otros, es decir, los recursos sólo podrán ser liberados voluntariamente por sus propietarios.

Espera circular: Si existen varios procesos P_1, P_2, \dots, P_n ; P_1 estará esperando por un recurso que tenga asignado p_2 . Idéntica situación se dará entre P_2 y P_3 , y así sucesivamente hasta P_n que esperará por uno que posea P_1 .

Recursos

Un sistema se compone de un numero finito de recursos que se distribuyen entre varios tipos:

Físicos: Ciclo de cpu, espacio en memoria, dispositivos de e/s (impresoras, unidades de cinta, etc.)

Lógicos: Ficheros, tablas del sistemas, semáforos.

Por lo general, una computadora tiene distintos recursos que pueden ser otorgados. Algunos recursos podrán tener varias instancias idénticas, como es el caso de tres unidades de cinta. Si se tienen disponibles varias copias de un recurso, cualquiera de ellas se puede utilizar para satisfacer cualquier solicitud del recurso. Un recurso es cualquier cosa que solo puede ser utilizada por un único proceso en un instante dado.

Recursos

Los recursos son de dos tipos:

- Apropiable

- No apropiables

Un recurso apropiable es aquel que se puede tomar del proceso que lo posee sin efectos dañinos. La memoria es un ejemplo de recurso apropiable.

Por el contrario, un recurso no apropiable, es aquel que no se puede tomar de su poseedor activo sin provocar un fallo de calculo. Si un proceso comienza a imprimir una salida, se toma la impresora y se le da a otro proceso, el resultado será una salida incomprensible. Las impresoras no son apropiables.

La secuencia de eventos necesaria para utilizar un recurso es:

- Solicitar el recurso
- Utilizar el recurso
- Liberar el recurso

1. Avestruz:

Banquero :

2. *Un recurso:*

3. *Varios recursos:*

Estados

Inseguro

Seguro

Niveles de Planificación

La planificación de la CPU es una función del SO.

Se lleva a cabo por el planificador a corto plazo o dispatcher(despachador).

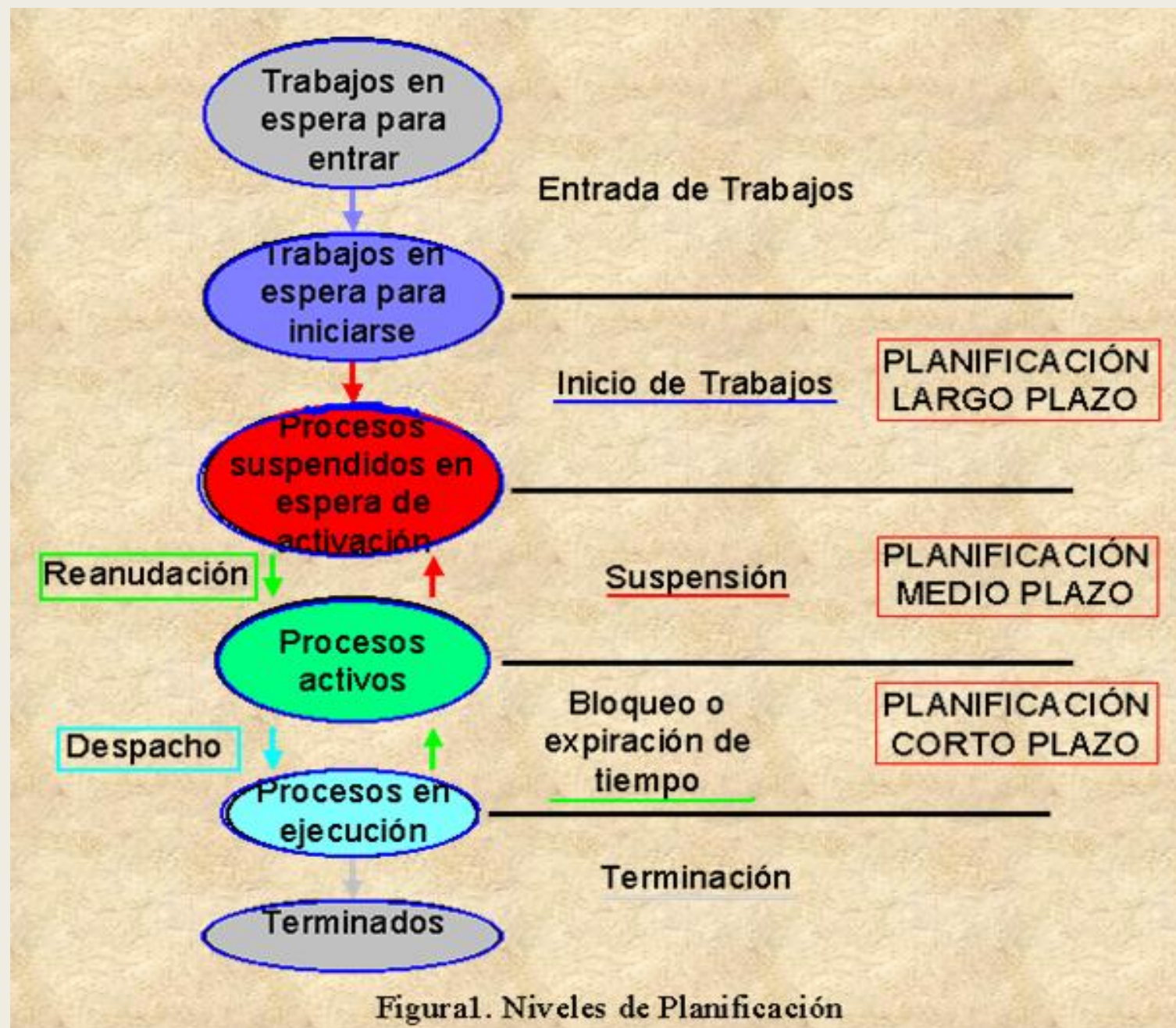
La misión del despachador consiste en asignar la CPU a uno de los procesos ejecutables del sistema

Los acontecimientos que pueden provocar la llamada al dispatcher dependen del sistema (son un subconjunto de las interrupciones)

Un elemento del sistema operativo ordena el bloqueo del proceso en ejecución.

EL proceso en ejecución agota su quantum de estancia en la CPU

Un proceso pasa a estado listo



Objetivos y criterios de planificación

El principal objetivo de la planificación a corto plazo es repartir el tiempo del procesador de forma que se optimicen algunos puntos del comportamiento del sistema. Generalmente se fija un conjunto de criterios con los que evaluar las diversas [estrategias de planificación](#). El criterio más empleado establece dos clasificaciones. En primer lugar, se puede hacer una distinción entre los criterios *orientados a los usuarios* y los *orientados al sistema*. Los criterios orientados al usuario se refieren al comportamiento del sistema tal y como lo perciben los usuarios o los procesos. Uno de los parámetros es el [tiempo de respuesta](#). El tiempo de respuesta es el periodo de tiempo transcurrido desde que se emite una solicitud hasta que la respuesta aparece en la salida. Sería conveniente disponer de una política de planificación que ofrezca un buen servicio a diversos usuarios.

Técnicas de Administración del Planificador

el planificador es un componente funcional muy importante de los sistemas operativos multitarea y multiproceso, y es esencial en los sistemas operativos de tiempo real. Su función consiste en repartir el tiempo disponible de un microprocesador entre todos los procesos que están disponibles para su ejecución".

FIFO

Guarda analogía con las personas que esperan en una cola y van siendo atendidas en el orden en que llegaron, es decir, que la primera persona que entra es la primera persona que sale".

Silbertschatz, C. (1999) menciona que "Los procesos se despachan de acuerdo con su tiempo de llegada a la cola de procesos listos. Cuando un proceso tiene la CPU, se ejecuta hasta terminar. Es justo en el sentido formal, pero algo injusta en cuanto a que los trabajos largos hacen esperar a los cortos y los trabajos sin importancia hacen esperar a los importantes".

Segun Silbertschatz, C. (1999). "este es un algoritmo que no usa apropiación, y que consiste en atender a los procesos por estricto orden de llegada a la lista de procesos listos".

FIFO

Silbertschatz, C. (1999) menciona "las características principales de este algoritmo son las siguientes:"

No es apropiativa.

Es justa, aunque los procesos largos hacen esperar mucho a los cortos.

Es una política predecible.

El tiempo promedio de servicio es muy variable ya que esta en función del número de procesos y la duración promedio que tenga.

SJF

(SJF, planificación con selección del trabajo más corto). Este algoritmo asocia con cada proceso la duración de la siguiente ráfaga de CPU del proceso. Cuando la CPU está disponible, se asigna al proceso que tiene la siguiente ráfaga de CPU más corta. Si las siguientes ráfagas de CPU de dos procesos son iguales, se usa la planificación FCFS para romper el empate.

RR

Silbertschatz, C. (1999), "los procesos se despachan en "FIFO" y disponen de una cantidad limitada de tiempo de cpu, llamada "división de tiempo" o "cuanto"(RR, Planificación de Asignación en Rueda)".

Silbertschatz, C. (1999), describe que cada proceso recibe una fracción de tiempo de procesamiento o cuanto para su ejecución, de manera que cuando se está ejecutando y excede el tiempo que se le ha concedido, se genera una interrupción de reloj, mediante la cual la ejecución del proceso se detiene y se coloca al proceso al final de la cola de procesos 'listos' para su posterior ejecución, seleccionándose a continuación un nuevo proceso de la cola para su ejecución. Si un proceso finaliza su ejecución antes de que termine el tiempo que se le ha asignado, este cede el control, seleccionándose un nuevo proceso de la cola para su ejecución.

Queves Multi-level

Según la investigación de Silbertschatz, C. (1999), un algoritmo de planificación mediante 'colas multinivel' divide la cola de procesos preparados en varias colas distintas. Los procesos se asignan permanentemente a una cola, generalmente en función de alguna propiedad del proceso, como por ejemplo el tamaño memoria, la prioridad del proceso o el tipo de proceso. Cada cola tiene su propio algoritmo de planificación. Por ejemplo, pueden emplearse colas distintas para los procesos de primer plano y de segundo plano. La cola de primer plano puede planificarse mediante un algoritmo por turnos, mientras que para la cola de segundo plano puede emplearse un algoritmo FCFS.