

Sistemas Operativos

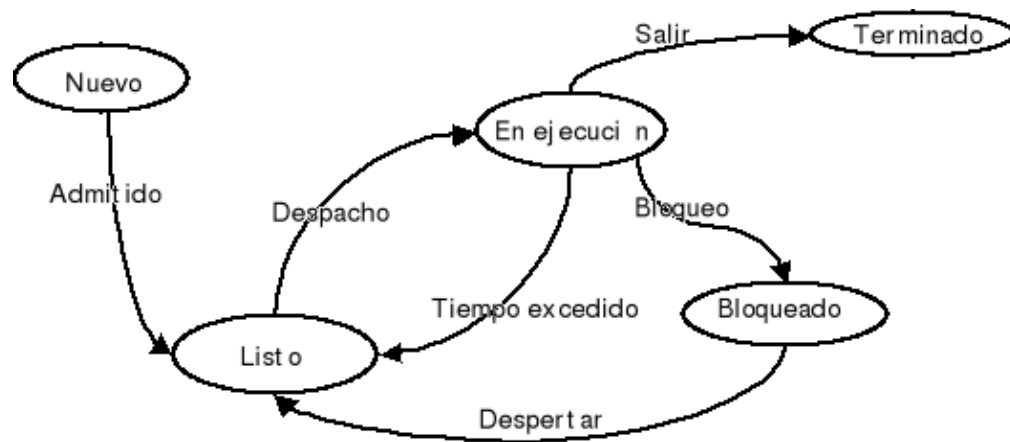
(Procesos)

Procesos

Def: Un proceso es un programa en ejecución.

Estados de procesos

El estado de un proceso define su actividad actual. Durante su existencia, un proceso pasa por una serie de estados discretos. Estos estados son se muestran en el siguiente diagrama:



A continuación se describe cada uno de estos procesos:

Estado	Descripción
<i>Nuevo</i>	Se dice que un proceso está en estado de nuevo cuando apenas se encuentra en proceso de crearse.
<i>Listo</i>	Un proceso está en estado de listo, cuando podría usar una UCP, si hubiera una disponible.
<i>En ejecución</i>	Se dice que un proceso está estado de ejecución, si en ese momento tiene esta ocupando la CPU.
<i>Bloqueado</i>	Se dice que un proceso está en estado de bloqueado, si espera que ocurra algo, como por ejemplo, la terminación de una E/S, para así poder ponerse en marcha.
<i>Terminado</i>	Cuando un proceso se ha completado su ejecución pasa a ser un proceso terminado.

Transiciones de estados:

Cuando un trabajo es admitido se crea un proceso equivalente, y es insertado en la ultima parte de la cola de listos (ésta se tratara más adelante). Cuando un proceso pasa de un estado a otro se dice que hace una transición de estado, estas transiciones se describen a continuación.

Transición	Descripción
Admitido(Proceso):Nuevo Listo	Cuando un proceso se ha creado y se le es permito para competir por la CPU.
Despacho(Proceso):Listo En ejecución	

	La asignación de la CPU al primer proceso de la lista de listos es llamado despacho y es ejecutado por la entidad de sistema llamada despachador. Mientras que el proceso tenga la CPU se dice que esta en ejecución.
Tiempo excedido(Proceso):En ejecución Listo	El S.O , cuando un proceso se le expira el intervalo de tiempo asignado para estar en ejecución (CUANTO), hace que este proceso que se hallaba en estado de ejecución pase al estado de listo y inmediatamente el despachador hace que el primer proceso de la lista pase a estado de ejecución.
Bloqueo(Proceso):En ejecución Bloqueado	Si un proceso que se encuentra en estado de ejecución inicia una operación de E/s antes que termine su cuanto, el proceso voluntariamente abandona la CPU, es decir, el proceso se bloquea a sí mismo.
Despertar(Proceso):Bloqueo Listo	La única transición posible en nuestro modelo básico ocurre cuando acaba una operación de E/S (o alguna otra causa por la que esté esperando el proceso), y esta termina pasa a el estado de listo.
Salir(Proceso):En ejecución Terminado	Esta transición ocurre cuando el proceso se ha terminado de ejecutarse, y pasa a un estado de terminado.

Nota: Para prevenir que un proceso monopolice el sistema, el S.O ajusta un reloj de interrupción del hardware para permitir al usuario ejecutar su proceso durante un intervalo de tiempo específico. Cuando este tiempo expira el reloj genera una interrupción, haciendo que el S.O recupere el control.

Cuando hay demasiada carga en el sistema se puede hacer uso de suspensión y reanudación por el S.O, para equilibrar la carga del sistema. Para la reanudación y la suspensión será necesario anexar otros dos estado los cuales son: *suspendido listo* y *suspendido bloqueado*, con las siguiente definición de transiciones:

Suspende_ejecución(Proceso): En ejecución Suspendido listo

Suspende_bloqueado(Proceso): Bloqueado Suspendido bloqueado

Reanuada(Proceso): Suspendido listo Listo

Termino_E/S(Proceso): Suspendido bloqueado Suspendido listo

Bloque de control de proceso (PCB):

La manifestación de un proceso en un S.O es un bloque de control de proceso (PCB), es decir cada proceso es representado por su PCB. El PCB es una estructura de datos que contiene cierta información importante acerca del proceso.

Esta información es:

- Estado actual del proceso
- Id único para proceso
- Prioridad del proceso
- Apuntadores para localizar la memoria del proceso
- Apuntadores para asignar recursos
- Area para preservar registros.

El PCB es un almacenamiento central de información que permite al S.O localizar toda la información clave sobre el proceso.

Operaciones con procesos:

- Crear un proceso
- Destruir un proceso
- Suspender un proceso
- Reanudar un proceso
- Cambiar la prioridad de un proceso
- Bloquear un proceso
- Despertar un proceso
- Despachar un proceso

La creación de un proceso implica varias operaciones, incluyendo:

- Dar nombre al proceso
- Registrarlo en la lista de sistema de procesos conocidos
- Determinar la prioridad inicial del proceso
- Crear el PCB
- Asignar recursos iniciales

Un proceso puede crear varios procesos nuevos, a través de una llamada al sistema de crear proceso, durante el curso de ejecución. El proceso creador se denomina proceso *padre* , y los nuevos procesos son los *hijos* de ese proceso. Cada uno de estos procesos nuevos puede a su vez crear otros procesos, formando un *árbol* de procesos.

Cuando un proceso crea un proceso nuevo, hay dos posibilidades en términos de ejecución:

- El padre sigue ejecutándose de forma concurrente con sus hijos.
- El padre espera hasta que algunos de sus hijos, o todos, han terminado.

También hay dos posibilidades en términos del espacio de direcciones del nuevo proceso:

- El proceso hijo es un duplicado del proceso padre.
- Se carga un programa en el proceso hijo.

La terminación de procesos

Un proceso acaba cuando termina de ejecutar su último enunciado y le pide al S.O que lo elimine utilizando la llamada al sistema **salir** (exit). En este momento, el proceso podría devolver datos(salidas) a su proceso padre (por medio de la llamada al sistema **esperar**). El S.O liberará los recursos del proceso, incluidos la memoria física y virtual, archivos abiertos y *buffers* de E/S.

Hay otras circunstancias por la que un proceso puede terminar, esta puede ser por la llamada al sistema **abortar**.

Un padre podría terminar la ejecución de uno de sus hijos por diversas razones, como:

- El hijo ha excedido en la utilización de algunos de los recursos que se le asignaron.

- La tarea que se asignó al hijo ya no es necesaria.
- El padre va a salir, y el sistema operativo no permite que un hijo continúe si su padre termina.

Planificadores:

En un sistema por lotes cuando hay demasiados procesos de los que se pueden inmediatamente ejecutarse se deben de colocar en *spool* en un dispositivo de almacenamiento masivo (por lo regular un disco), y ahí esperar hasta que puedan ser ejecutados.

Para la planificación se hace uso de los sig. planificadores de procesos:

Planificador a corto plazo (o planificador de trabajos): este escoge procesos que estén en el spool (en disco) y los carga en la memoria para que se ejecuten.

Planificador a largo plazo (o planificador de CPU): este escoge entre los procesos que estén listos para ejecutarse, y asigna la CPU a uno de ellos.

El planificador a largo plazo se ejecuta con mayor frecuencia que el de corto plazo y también controla el grado de multiprogramación (numero de procesos que están en la MP).

Un proceso se describen como:

- Proceso limitado por CPU, este ocupa su mayor tiempo para realizar cálculos.
- Proceso limitado por E/S, este dedica su mayor tiempo a realizar operaciones de E/S.

En un sistema de tiempo se introduce un planificador más , el cual es:

Planificador a mediano plazo: este intercambia procesos a disco y luego intercambia a la memoria, a este esquema se le denomina intercambio (swapping). Esto puede ser necesario para mejorar la mezcla de procesos o cuando un cambio en las necesidades de memoria ha distribuido la memoria disponible entre demasiados procesos y es preciso liberar algo de memoria.

Comunicación entre procesos:

Hay dos formas en que se puede comunicar los procesos, los cuales son:

- Por medio de un esquema de comunicación por memoria compartida (Buffer)
- Por medio de un mecanismo de comunicación entre procesos (IPC, Interprocess communication).

La IPC ofrece un mecanismo que permite a los procesos comunicarse y sincronizar sus acciones. La mejor forma de proveer la comunicación entre procesos es mediante un sistema de mensajes.

La función de un sistema de mensaje es permitir a los procesos comunicarse entre sí sin tener que recurrir a variables compartidas.

Un recurso IPC ofrece por los menos 2 operaciones:

enviar (mensaje) (**send**) y **recibir** (mensaje) (**receive**).

Sea P y Q dos procesos que requieren comunicarse deberán enviarse mensajes; para ello debe existir un

enlace de comunicación entre ellos. Este enlace puede implementarse de diversas maneras.

Los métodos para implementar lógicamente un enlace y las operaciones de **enviar** / **recibir** son:

- Comunicación directa o indirecta
- Uso de buffer automático o explícito
- Envío por copia o envío por referencia
- Mensajes de tamaño fijo o variables

Comunicación directa:

Aquí cada proceso que desee comunicarse debe nombrar explícitamente el destinatario o el remitente de la comunicación. Este esquema se define las primitivas de la sig. manera:

Enviar(P,mensaje): Enviar un mensaje al proceso P.

Recibir(Q, Mensaje): Recibir un mensaje del proceso Q.

Con las siguientes propiedades:

- Se establece automáticamente el enlace entre cada par de procesos. Los procesos sólo necesitan conocer la identidad de otro para la comunicación.
- Solo hay un enlace entre cada par de procesos.
- El enlace puede ser unidireccional o bidireccional.

Este esquema exhibe una simetría de direccionamiento; es decir, los procesos tanto emisor como receptor necesitan nombrar al otro para comunicarse.

Otra variante de este esquema es utilizar asimetría de direccionamiento, con las sig. primitivas:

Enviar(P,mensaje): enviar un mensaje al proceso P.

Recibir(Id,mensaje) : recibir un mensaje de cualquier proceso con el que hubo comunicación.

Aquí sólo el emisor nombra al destinatario; el destinatario no está obligado a nombrar al emisor.

Comunicación indirecta:

Aquí los mensajes se envían a, y se reciben de, **buzones** (también llamados PUERTOS). Un buzón puede considerarse en lo abstracto como un objeto en el que los procesos pueden colocar mensajes y del cual se pueden sacar mensajes. Cada buzón tiene una identificación única. Aquí dos procesos se pueden comunicar sólo si comparten un buzón. Las primitivas se definen como:

Enviar (A,mensaje): enviar un mensaje al buzón A.

Recibir (A,mensaje): recibir un mensaje del buzón A.

Un enlace de comunicación tiene las sig. propiedades:

- Se establece un enlace entre un par de procesos sólo si tienen un buzón compartido.
- Un enlace puede estar asociado a más de dos procesos.
- Entre cada par de procesos en comunicación puede haber varios enlaces distintos, cada uno de los

cuales corresponderá a un buzón.

- Los enlaces pueden ser unidireccionales o bidireccionales.

Hay varias formas de designar el dueño de y los usuarios de un buzón dado. Una posibilidad es permitir que un proceso declare variables de tipo buzón. El proceso que declara un buzón es el dueño de ese buzón. Cualquier otro proceso que conozca el nombre de dicho buzón podrá usarlo.

Por otro lado, un buzón propiedad del S.O tiene existencia propia; es independiente y no está unido a ningún proceso específico. El S.O establece un mecanismo que permite a un proceso:

- Crear un buzón nuevo
- Enviar y recibir mensajes a través del buzón
- Destruir un buzón.

