



INDICE	PÁGINAS
⇒ INTRODUCCIÓN	86
⇒ TABLAS DE CONTROL DE UN SISTEMA OPERATIVO	87
⇒ MEMORIA – E/S – ARCHIVOS – PROCESOS.	
⇒ ESTADOS DE UN PROCESO	88
⇒ BLOQUE DE CONTROL DE PROCESOS	89
⇒ PLANIFICACIÓN DE PROCESOS	91
⇒ CRITERIOS DE PLANIFICACIÓN Y RENDIMIENTO	94
⇒ COMUNICACIÓN ENTRE PROCESOS	95
⇒ TIPOS DE COMUNICACIÓN	96
⇒ VARIANTES	97
⇒ CARACTERÍSTICAS	98
⇒ SERVICIOS DEL SISTEMA OPERATIVO PARA LA GESTIÓN DE PROCESOS	99
⇒ LLAMADAS DEL SISTEMA	101



PROCESOS

*“Nunca dejemos que las palabras dominen al pensamiento,
primero pensamos, después hablamos”.*

Daniel Slavutsky

INTRODUCCIÓN

Cada programa que es ejecutado, el sistema operativo debe primero transformarlo en un proceso. Un proceso es una unidad ejecutable, a la que el sistema operativo le ha asignado recursos tales como memoria principal, memoria secundaria, dispositivos periféricos y CPU.

Desde el punto de vista del sistema operativo, **un proceso es la entidad mínima individualmente planificable y que consta de códigos y datos y que se caracteriza por sus atributos y estado dinámico.**

Los códigos se componen de instrucciones de máquina y de llamadas de servicio del sistema operativo. Los atributos los puede asignar el programador, o el sistema operativo mismo puede incluir como atributos el orden prioridades para los procesos, y el derecho de acceso a los diferentes recursos.

A medida que se ejecuta el proceso cambia de estado. El estado de un proceso se define por su actividad. La ejecución de un proceso es una secuencia alternante de ciclos de CPU y de E/S, terminando siempre por un ciclo de CPU. Un proceso puede encontrarse en uno de los siguientes estados: inactivo, preparado, en ejecución y en suspenso.

Al conjunto de uno o más procesos se lo define como a un trabajo, por ejemplo, si se tiene que correr el inventario, tal vez se subdivida ese trabajo en varios procesos: obtener la lista de artículos, número en existencia, artículos vendidos, artículos extraviados, etc.

A los subprocesos o sección que genera a partir de un proceso se lo llama hilos (thread), que reunirá las mismas condiciones que tiene todo proceso, dado que también deberá ser planificado y por consecuencia podrá encontrarse en cualquiera de sus estados, la única diferencia que para terminar su ejecución de forma normal deberá volver a su proceso origen.

El **grado de granularidad** que posea un sistema operativo estará asociado a la **cantidad de trabajos, procesos e hilos de un proceso** que deba realizar con uno o varios procesadores.

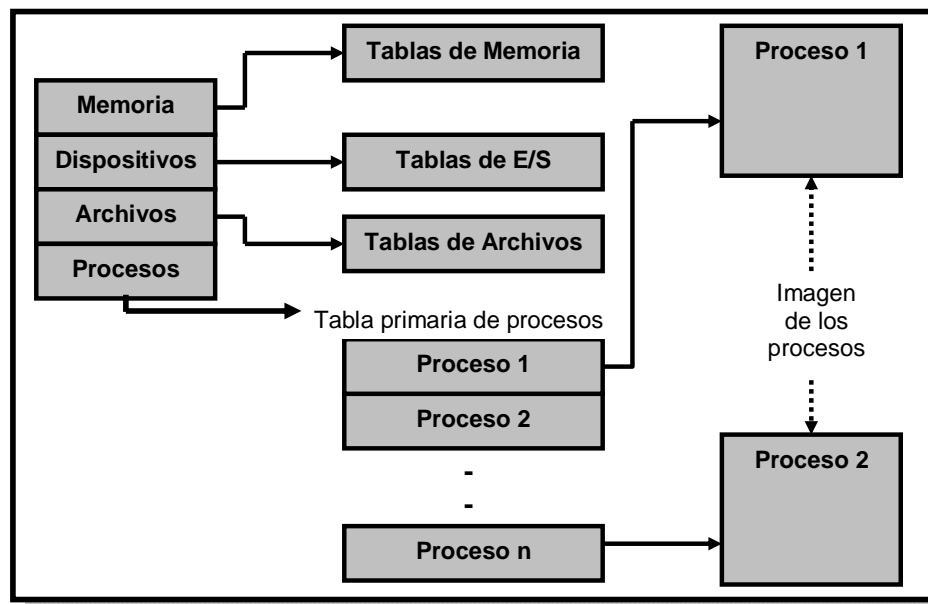


TABLAS DE CONTROL DE UN SISTEMA OPERATIVO

El sistema operativo utiliza para la administración de los procesos distintas tablas que le proporciona información sobre la entidad que va a gestionar. Las tablas que utiliza son de memoria, E/S, archivos y procesos.

- **Tabla de Memoria:** Se utilizan para mantener un registro tanto de la memoria principal (real) como de la secundaria (virtual). Esta tabla incluye la siguiente información: espacio de memoria principal reservada por parte de los procesos, espacio de memoria secundaria, información para la protección de los espacios asignados e información para el manejo de la memoria virtual.
- **Tabla de E/S:** Sirve para gestionar a los dispositivos de E/S conteniendo información sobre el estado en que se encuentra (en uso o no), dirección de la memoria principal usada como origen de los datos o la dirección del destino de la transferencia de E/S.
- **Tabla de Archivos:** Proporcionan sobre la existencia de los archivos, su ubicación en la memoria secundaria, su estado actual y otros atributos. Esta tabla cumple funciones similares a la del sistema de archivos, pero a diferencia esta tabla a diferencia del sistema de archivos puede ser accedida a modo usuario.
- **Tabla de Procesos:** Sirven para gestionar a los procesos y posee información que le brindan como referencia las otras tablas, dado que directa o indirectamente son gestionada desde esta tabla.

Tablas de control del sistema operativo¹

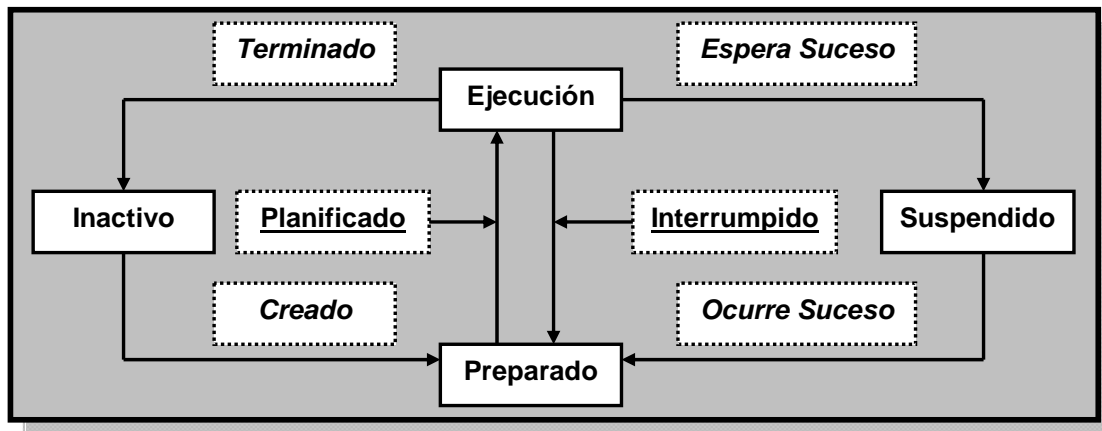


ESTADOS DE UN PROCESO

¹ William Stallings, Sistemas Operativos. Aspectos internos y principios de diseño. Página 127, Editorial, Pearson Educación S. A., 2005.



- **Inactivo:** Porque indica que un proceso no es conocido por el sistema operativo. Todos los procesos en espera de activación, así como los programas no presentados al sistema operativo, pueden ser considerados como inactivos.
- **Preparado:** El proceso preparado posee todos los recursos que se necesitan para su ejecución, a excepción de la CPU. Todos los procesos preparados están esperando a que el sistema operativo le asigne la CPU para activarse. Un módulo del sistema operativo al que se denomina **planificador**, selecciona a un proceso preparado para su ejecución, siempre que el sistema operativo tenga el control de la CPU, siempre estará a punto de devolver el control de ésta a un nuevo proceso.
- **Ejecución:** El proceso en ejecución posee todos los recursos necesarios para su ejecución, incluida la CPU. El proceso en ejecución ejecuta su frecuencia de instrucciones de máquina y puede recurrir al sistema operativo para que preste servicios tales como la E/S de datos a la sincronización mediante el intercambio de señales en su nombre. Según la política de planificación que este en vigor, el sistema operativo puede devolver el control al proceso en ejecución, una vez prestado el servicio, o bien planificar otro proceso si hay uno preparado para su ejecución.
- **Suspensión:** Al proceso en **suspensión** le falta algún recurso aparte de la CPU, normalmente, a dichos procesos se les deja fuera de competición por conseguir su ejecución, hasta que se le elimina su condición de suspendido. Un proceso en ejecución puede quedar en suspenso al invocarse una rutina de E/S cuyos resultados necesita para seguir adelante, o bien, al esperar una señal que aún no se ha producido.



Los estados colectivos de todos los procesos y recursos (ocupados o libres) del sistema pueden ser considerados como el **estado global del sistema**. En respuesta a acontecimientos tanto internos o externos, los procesos pueden cambiar rápidamente de estado y formar un estado global del sistema.

El sistema operativo agrupa toda la información que necesita acerca de un proceso determinado, en una estructura de datos llamada **bloque de control de proceso**² (BCP).

² Peterson, J. L. – Silberschatz, A. (1989); *Sistemas Operativos. Conceptos Generales: Segunda Edición*; Editorial Reverté, S.A.; Barcelona, España; página 116.



El bloque de control de proceso (BCP), sirve simplemente como depósito de cualquier información que pueda variar de proceso a proceso. Los BCP se almacenan en la parte de la memoria ocupada por el componente del sistema operativo llamado monitor y existirán tantos BCP como procesos estén preparados para ejecutarse.

BLOQUE DE CONTROL DE PROCESOS³
Identificación del Proceso
Los identificadores son datos numéricos que se pueden guardar en un bloque de control de procesos. Tipos: <ul style="list-style-type: none">• Identificador de este proceso.• Identificador del proceso que creo a este procesos (el proceso padre).• Identificador del usuario.
Información visibles para el usuario
Registros visibles para el usuario: Un registro visible para el usuario es aquél al que puede hacerse referencia por medio del lenguaje de máquina que ejecuta el procesador. Normalmente, existen de 8 a 32 de estos registros, aunque algunas implementaciones RISC tienen más de 100.
Registros de control y de estado: Hay varios registros del procesador que se emplean para controlar su funcionamiento. Entre estos se incluyen: <ul style="list-style-type: none">• Contador del Programa: contiene la dirección de la próxima instrucción a ejecutar.• Códigos de condición: muestran el resultado de la operación aritmética o lógica más reciente (signo, cero, acarreo, igualdad, desbordamiento).• Información de estado: incluye los indicadores de habilitación o inhabilitación de interrupciones y el modo de ejecución.
Punteros de pila: Cada proceso tiene una o más pila LIFO del sistema asociadas. Las pilas se utilizan para almacenar los parámetros y las direcciones de retorno de los procedimientos y de las llamadas al sistema. El puntero de pila siempre apunta a la cima de la pila.
Información de control del proceso
Información de planificación y de estado: esta es la información que necesita el sistema operativo para llevar a cabo sus funciones de planificación de los procesos. <ul style="list-style-type: none">• Estado del proceso: define la disposición del proceso para ser planificado para ejecutar (en ejecución, preparado, suspendido).

³ William Stallings, Sistemas Operativos. Aspectos internos y principios de diseño. Página 130, Editorial, Pearson Educación S. A., 2005.



- **Prioridad:** se puede usar uno o más campos para describir la prioridad de planificación de los procesos. En algunos sistemas se necesitan varios valores (por omisión, actual, la más alta permitida).
- **Información de planificación:** esta dependerá del algoritmo de planificación utilizado. Como ejemplos se tienen la cantidad de tiempo que el proceso ha estado esperando y la cantidad de tiempo que el proceso se ejecutó la última vez.
- **Suceso:** la identidad del suceso que el proceso esta esperando antes de poder reanudarse.

Estructuras de datos: un proceso puede estar enlazado con otros procesos en una cola de espera o algún tipo de estructura. Por ejemplo varios procesos pueden estar enlazados en el estado de preparados en un nivel determinado de prioridades en una cola de espera.

Un proceso puede mostrar una relación **padre-hijo (creador-creado)** con otro proceso. Un bloque de control de procesos puede contener punteros a otros procesos para servir de soporte de estas estructuras.

Comunicación entre procesos: pueden haber varios indicadores, señales y mensajes asociados con la comunicación entre dos procesos independientes. Una parte de esta información o toda ella se puede guardar en el bloque de control de procesos.

Privilegios de los procesos: a los procesos se les otorgan privilegios en términos de la memoria a la que pueden acceder y el tipo de instrucciones que pueden ejecutar. Además, también pueden aplicar privilegios al uso de los servicios y utilidades del sistema.

Gestión de memoria: esta sección puede incluir punteros a las tablas de páginas o segmentos que describen la memoria virtual asignada a un proceso.

Propiedad de los recursos y utilización: se pueden indicar los recursos controlados por el proceso, como los archivos abiertos. También se puede incluir un historial de la utilización del procesador o de otros recursos, esta información puede ser necesaria para el procesador.

PLANIFICACION DE PROCESOS

La planificación es una función fundamental del sistema operativo, puesto que todos los recursos del computador se planifican antes de utilizarlos. La CPU es el recurso principal y por lo tanto su planificación es algo esencial en el diseño del sistema operativo.



La CPU ejecuta un gran número de trabajos o tareas. Cada acción del sistema es iniciado por la CPU. La CPU tiene que responder a excepciones de error, a las llamadas de los programas e interrupciones de E/S.

El éxito de la planificación de la CPU depende de las siguientes propiedades: la ejecución de un proceso es un **ciclo de ejecución** en la CPU y espera para E/S.

Los procesos alternan estos dos estados. Un proceso empieza con un ciclo de ejecución de la CPU, le sigue un ciclo de E/S, seguida por otro ciclo de CPU, y luego por otro ciclo de E/S, etc. Finalmente, se ejecutará otro ciclo de CPU que terminará con una llamada al sistema para la terminación del programa. Nunca termina un programa con un ciclo de E/S.

Un sistema operativo tiene muchos planificadores. Hay tres tipos de planificadores para la CPU (largo plazo, corto plazo y mediano plazo) y uno para los dispositivos de E/S.

El **planificador a largo plazo** o **planificador de procesos**, determina cuales son los procesos que admite el sistema para su procesamiento, los transfiere a una unidad de almacenamiento, y luego los carga en la memoria para su posterior ejecución. Por cada programa admitido crea un nuevo proceso, agregándolo a la cola de procesos preparados para ejecutarse.

El **planificador a corto plazo** o **planificador de la CPU** selecciona entre los procesos que estén en memoria viendo cuál está listo para ejecutarse asignándole el uso de la CPU.

El planificador a corto plazo, también se lo conoce como distribuidor (dispatcher), es el de ejecución más frecuente y toma decisiones con un mayor detalle sobre el proceso que se ejecutará a continuación.

Este planificador entra en actividad cuando ocurre un suceso que pueda conducir a la interrupción del proceso actual o que ofrece la oportunidad de hacerlo a favor de otro proceso que esté esperando su ejecución. Como ejemplos de sucesos que pueden ocurrir durante la ejecución de un proceso podemos mencionar:

- Interrupciones del reloj.
- Interrupciones de E/S.
- Llamadas al sistema operativo.
- Señales.

La distinción fundamental entre estos dos planificadores es la frecuencia en la cuál se ejecutan los procesos. El planificador a corto plazo tiene que seleccionar un nuevo proceso para la CPU con mucha frecuencia. Un proceso se ejecuta durante sólo unos pocos milisegundos antes de tener que esperar por haber realizado una llamada de E/S. El planificador a largo plazo se ejecuta con mucha menor frecuencia, pueden pasar varios minutos entre la llegada de nuevos trabajos al sistema. El planificador a largo plazo controla el **grado de multiprogramación**, o sea, el número de procesos en memoria.

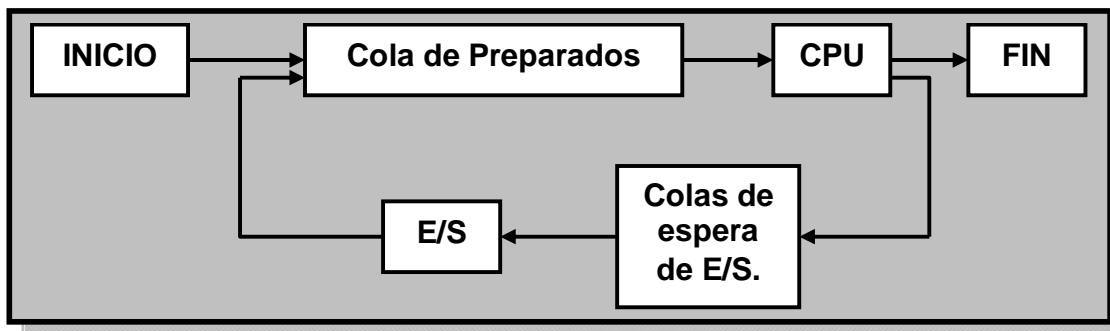


Algunos sistemas, especialmente aquellos que tienen memoria virtual o tiempo compartido, introducen un nivel intermedio de planificación a **mediano plazo** que en algunos casos puede resultar ventajoso eliminar procesos de memoria y reducir así el grado de multiprogramación. Más tarde el proceso puede reintroducirse y continuar en el punto donde lo abandonó. Este tipo de planificador utiliza frecuentemente una técnica de intercambio de espacio en memoria (real con virtual) que se denomina swapping, o sea, que el planificador a mediano plazo cambia un proceso haciéndolo salir de la memoria RAM volviéndolo a cargar mas tarde.

Todo proceso que preparado y esperando su ejecución se mantienen en una lista llamada **cola de preparados**. Espera en ella hasta que es seleccionado por la CPU. Después de ejecutarse en la CPU espera una operación de E/S pasando a una cola de E/S y finalmente recibe servicio del dispositivo de E/S y vuelve a la cola de preparados. El proceso continúa este ciclo de CPU y E/S hasta que termine y sale del sistema.

El **planificador de E/S** ⁴ esta asociado a las solicitudes pendientes por parte de un proceso para el uso de alguno de los dispositivos de E/S que intervengan durante la vida de un proceso.

Ejemplo de un diagrama de cola:⁵



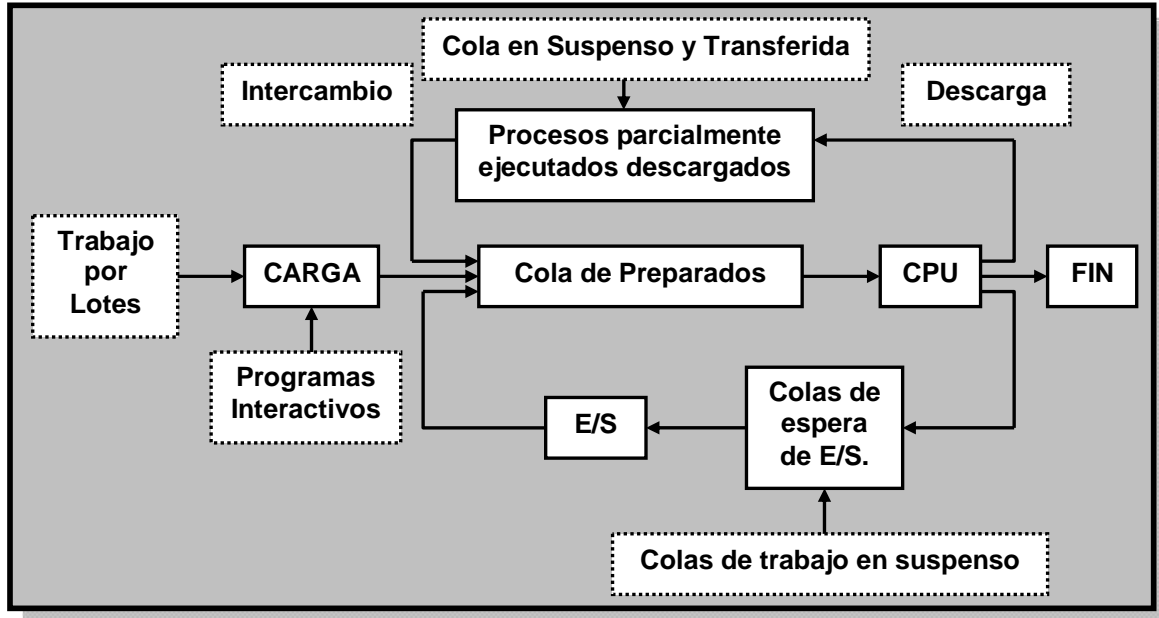
- En la cola de preparados se encuentran todos los procesos que han sido reconocidos por el sistema operativo pero que están esperando la asignación de la CPU por parte del componente Scheduler o administrador de procesos a ser ejecutados.
- En la cola de espera de E/S se encuentran los procesos que están esperando por algún dispositivo de E/S que esta siendo ocupado por otro proceso y espera de que le sea asignado por parte del sistema operativo.

Ejemplo gráfico de planificación:⁶

⁴ William Stallings, Sistemas Operativos. Aspectos internos y principios de diseño. Página 130, Editorial, Pearson Educación S. A., 2005.

⁵ Peterson, J. L. – Silberschatz, A. (1989); *Sistemas Operativos. Conceptos Generales: Segunda Edición*; Editorial Reverté, S.A.; Barcelona, España; página 121.

⁶ Peterson, J. L. – Silberschatz, A. (1989); *Sistemas Operativos. Conceptos Generales: Segunda Edición*; Editorial Reverté, S.A.; Barcelona, España; página 121.



- Procesos parcialmente ejecutados descargados corresponden al módulo de un planificador a mediano plazo.
- La cola de preparados corresponde a un planificador a corto plazo.
- El proceso desde su carga y su final corresponde al planificador a largo plazo.

CRITERIOS DE PLANIFICACION Y RENDIMIENTO⁷

Se han sugerido muchos criterios para la planificación de la CPU para optimizar el rendimiento del sistema. Entre los criterios que se emplean están los siguientes:

1. Criterios orientados al usuario:

- 1.1. **Tiempo de retorno:** Es el intervalo de tiempo transcurrido entre el inicio de un proceso y su finalización. Es la suma del tiempo de ejecución real y el tiempo consumido en la espera por los recursos, incluido el procesador.

⁷ William Stallings, Sistemas Operativos. Aspectos internos y principios de diseño. Página 130, Editorial, Pearson Educación S. A., 2005.



- 1.2. **Tiempo de espera:** Podríamos considerar como tiempo de espera a la cantidad de tiempo que un trabajo invierte en esperar en la cola de preparados para su ejecución.
- 1.3. **Tiempo de respuesta:** Tiempo de respuesta es el tiempo que se tarda en iniciar la respuesta, no en el tiempo que lleva emitir esta respuesta. Otra medida es el tiempo desde el envío de una solicitud hasta que se produzca la primera solicitud.
- 1.4. **Plazos:** Cuando se pueden especificar plazos de terminación de un proceso, se puede maximizar el porcentaje de los plazos cumplidos.
- 1.5. **Previsibilidad:** Un determinado trabajo se debe ejecutar en el mismo tiempo y con el mismo costo sin importar la carga del sistema. Puede indicar una descompensación importante en la carga del sistema que indique la necesidad de remediar el inconveniente.

2. Criterios orientados al sistema de computación:

- 2.1. **Productividad:** Si la CPU se mantiene ocupada entonces se está realizando un trabajo. Una medida de este trabajo es la productividad o números de trabajos que se completan por unidad de tiempo. Se debe tener en cuenta para la planificación el tiempo medio de ejecución de cada proceso.
- 2.2. **Utilización de la CPU:** Es el porcentaje de tiempo en el que el procesador está ocupado, en un sistema de tiempo compartido (multiusuario), éste es un criterio muy importante. En cambio en los sistemas del tipo monousuario o de tiempo real, éste criterio tiene menor importancia.
- 2.3. **Equidad:** En ausencia de órdenes por parte del usuario o del sistema que obliguen a una tarea determinada, los procesos deben ser tratados de la misma forma y ningún proceso debe ser suspendido si no existe algún suceso que lo ocasione.
- 2.4. **Prioridades:** Cuando se le asignan prioridades a los procesos, la política de planificación debe favorecer a los de mayor prioridad.
- 2.5. **Equilibrio de recursos:** La política de planificación debe mantener ocupados los recursos del sistema, sin que ningún proceso monopolice en el uso de alguno de ellos. Este criterio puede afectar a cualquiera de los planificadores.

COMUNICACIÓN ENTRE PROCESOS

La comunicación entre procesos, en inglés IPC (Inter Process Communication) es una función básica de los sistemas operativos. Los procesos pueden comunicarse entre sí a través de compartir espacios de memoria, ya sean variables compartidas o buffers, o a través de las herramientas provistas por las rutinas de IPC.



La IPC provee un mecanismo que permite a los procesos comunicarse y sincronizarse entre sí, normalmente a través de un sistema de bajo nivel de paso de mensajes que ofrece la red subyacente.

La comunicación se establece siguiendo una serie de reglas (protocolos de comunicación). Los protocolos desarrollados para internet son los mayormente usados y están basados en el modelo de referencia OSI : IP (capa de red), protocolo de control de transmisión (capa de transporte) y protocolo de transferencia de archivos , protocolo de transferencia de hipertexto (capa de aplicación)⁸.

Los procesos pueden estar ejecutándose en una o más computadoras conectadas a una red. Las técnicas de IPC están divididas dentro de métodos para: paso de mensajes, sincronización, memoria compartida y llamadas de procedimientos remotos (RPC). El método de IPC usado puede variar dependiendo del ancho de banda y latencia (el tiempo desde el pedido de información y el comienzo del envío de la misma) de la comunicación entre procesos, y del tipo de datos que están siendo comunicados.

Un sistema operativo provee mínimamente dos operaciones primitivas, la de enviar y recibir, normalmente llamadas **send** y **receive**. Asimismo, debe implementarse un enlace de comunicación entre los procesos de la comunicación. Este enlace puede ser unidireccional o multidireccional según permita la comunicación en solo uno o en varios sentidos.

⁸ Ver SO-ANEXO VI-SISTEMAS MULTIUSUARIOS.



TIPOS DE COMUNICACIÓN

- **Comunicación Directa:** Cada proceso que desea enviar o recibir un mensaje tiene que nombrar claramente al receptor o al remitente de la comunicación.

Propiedades:

- a) Se establece un enlace automático entre cada par de procesos que desean comunicar. Los procesos sólo necesitan saber la identidad del otro para poder comunicarse.
- b) Un enlace está asociado exactamente a dos procesos.
- c) Para cada par de procesos en comunicación, hay exactamente un enlace.
- d) El enlace es bidireccional.

Este esquema es **simétrico** en el direccionamiento; es decir, tanto el que envía como el que recibe tiene que nombrar al otro para comunicar. Una variante es **asimétrica** en el direccionamiento. Únicamente el remitente nombra al receptor; el receptor no es requerido para que nombre al remitente.

- **Comunicación Indirecta:** En la comunicación indirecta, los mensajes se envían y reciben de **buzones** (también conocidos como **ports**). Un buzón puede definirse como un objeto en el que los procesos pueden situar mensajes, y del que pueden extraer mensajes. Cada buzón tiene una identificación única que lo distingue. Dos procesos se pueden comunicar solamente si comparten un buzón.

Propiedades:

- a) Se establece un enlace entre un par de procesos, solamente si comparten un buzón.
- b) Un enlace puede estar asociado con más de dos procesos.
- c) Entre cada par de procesos que comunican puede haber diferentes enlaces, correspondientes cada uno de ellos a un buzón.
- d) El enlace puede ser unidireccional o bidireccional.

Propiedades de los buzones: Un buzón puede ser propiedad de un proceso o bien del sistema operativo. Si el buzón es propiedad de un proceso, esto quiere decir que el buzón está asociado o definido como parte de un proceso, de esta forma distinguimos entre el



propietario (que a través de este buzón solamente puede recibir mensajes) y el usuario del buzón (que a través del mismo sólo puede recibir mensajes). Por otra parte, un buzón que es propiedad del sistema operativo, existe por sí mismo. Es independiente y no está asociado a ningún proceso.

El sistema operativo facilita un mecanismo que le permite a un proceso:

- 1) Crear un nuevo buzón: Enviar y recibir mensajes a través del buzón.
- 2) Destruir un buzón.

Mensajes: Los mensajes enviados a un proceso pueden ser de 3 tipos:

- **De tamaño fijo:** Si sólo pueden enviarse este tipo de mensajes, la implementación física es inmediata. Sin embargo esta restricción hace más difícil la tarea de programación.
- **De tamaño variable:** Este tipo de mensajes requiere una implementación física más compleja, pero la programación es más sencilla.
- **Mensajes con tipo:** La asociación de un tipo con cada buzón, es aplicable básicamente sólo a la comunicación indirecta. Los mensajes que pueden enviarse a y recibirse de un buzón están restringidos al tipo designado.

VARIANTES

- **Síncrona:** Quien envía permanece bloqueado esperando a que llegue una respuesta del receptor antes de realizar cualquier otra tarea.
- **Asíncrona:** Quien envía continúa con su ejecución inmediatamente después de enviar el mensaje al receptor.
- **Persistente:** El receptor no tiene que estar operativo al mismo tiempo que se realiza la comunicación, el mensaje se almacena tanto tiempo como sea necesario para poder ser entregado (Ej.: e-Mail).
- **Momentánea:** El mensaje se descarta si el receptor no está operativo al tiempo que se realiza la comunicación. Por lo tanto no será entregado.
- **Simétrica:** Todos los procesos pueden enviar o recibir. También llamada bidireccional para el caso de dos procesos.
- **Asimétrica:** Un proceso puede enviar, los demás procesos solo reciben. También llamada unidireccional.

CARACTERÍSTICAS



Los distintos procesos dentro de una computadora no actúan, evidentemente aislados. Por un lado deben **cooperar** con el fin de alcanzar el objetivo de poder ejecutar las tareas de los usuarios y por el otro, **compiten** por el uso de los diferentes recursos.

Estas dos actividades tienen asociada la necesidad de algún tipo de comunicación, la cual puede funcionar correctamente o incorrectamente planteándose de esta manera las siguientes características:

1. **Exclusión Mutua:** Existen recursos en un sistema que son compartibles y pueden ser usados por varios procesos en forma concurrente y otros no compartibles, o sea que su uso se ve restringido a un solo proceso por vez.

El hecho de no ser compatible puede deberse a cuestiones de imposibilidad técnica o por la razón de que es necesario su uso individual en virtud de que algún tipo de proceso puede interferir con el que lo está usando.

El problema que trata la exclusión mutua es el de asegurar que los recursos no compartibles sean accedidos por un solo proceso a la vez.

2. **Sincronización:** En términos generales la velocidad de un proceso respecto a otro es impredecible ya que depende de la frecuencia de la interrupción asociada a cada uno de ellos y de cuantas veces y por cuanto tiempo tuvo asignado el recurso del procesador. Se dice entonces, que un proceso se ejecuta asincrónicamente respecto de otro.

Sin embargo existen ciertos instantes en los cuales los procesos deben sincronizar sus actividades. Son estos puntos a partir de los cuales un proceso no puede continuar hasta tanto se haya completado algún tipo de actividad. El sistema operativo debe proveer los mecanismos que permitan poder llevar a cabo esta sincronización.

3. **Abrazo Mortal (Deadlock):** Cuando varios procesos compiten por los recursos, es posible que exista una situación en la cual ninguno de ellos puede proseguir debido a que los recursos que necesita están ocupados por otro proceso. Esta situación se conoce con el nombre de abrazo mortal. Evitarlos o limitar sus efectos, es claramente una de las funciones de los sistemas operativos.

SERVICIOS DEL SISTEMA OPERATIVO PARA LA GESTION DE PROCESOS



Los servicios en la ejecución de un proceso, son las llamadas del sistema que son invocadas por los procesos de usuarios.

Estas llamadas al monitor pueden estar incorporadas en el código del proceso, o bien, mediante ordenes dadas desde un puesto de trabajo, para luego ser traducidas por el Sistema Operativo.

Los sistemas de multiprogramación difieren frecuentemente en su filosofía de diseño y objetivos, pero los niveles que pertenecen al monitor (núcleo) presentan una gran similitud, aunque los parámetros de un sistema a otro varíen inevitablemente, las funciones que desempeña en el conjunto total de llamadas al sistema operativo son muy similares, las funciones y servicios del sistema operativo nos permitirán soportar la gestión de procesos para un entorno de multiproceso o proceso único. Las llamadas más importantes al sistema operativo son: CREATE, DELETE, FORK / JOIN, ABORT, SUSPEND, RESUME, DELAY, GET ATTRIBUTES Y CHANGE PRIORITY.

- **CREATE** (Creación): En respuesta a esta llamada, el sistema operativo crea un proceso nuevo con los atributos especificados por defecto y un identificador.

Un proceso no puede crearse a sí mismo, porque tendría que estar ejecutándose para invocar al sistema operativo y no se lo puede ejecutar antes de crearlo. Por ese motivo el proceso debe ser creado por otro proceso, nunca se puede crear a sí mismo.

En respuesta a la llamada CREATE, el sistema operativo obtiene un BCP de la memoria disponible, rellena los campos con los parámetros proporcionados por defecto e incluye el BCP en la lista de procesos preparados. Algunos de los parámetros que se pueden definir en el momento de la creación del proceso, comprenden:

1. Nivel de privilegio, tal como sistema o usuario.
 2. Prioridad.
 3. Tamaño y necesidades de memoria.
 4. Área máxima de memoria.
 5. Derechos de acceso e información sobre protección de memoria.
 6. Otros datos dependientes del sistema.
-
- **DELETE** (Eliminación): La invocación de esta llamada hace que el sistema operativo elimine del sistema el proceso designado. A diferencia de CREATE, un proceso puede eliminarse a sí mismo o a otro. El sistema operativo reacciona reclamando todos los recursos asignados al proceso especificado, cerrando los archivos abiertos por o para el proceso y realizando todas las demás operaciones de preparación inicial que sean necesarias. A continuación de este proceso, el BCP es eliminado de la lista en la que



reside en ese momento y se le hace volver al fondo común libre de memoria, lo cual, da lugar a que el proceso indicado quede inactivo. El servicio es invocado, normalmente, como parte de una terminación ordenada del programa.

- **FORK/JOIN** (Bifurcación/Concatenación): La operación FORK se utiliza para dividir una secuencia de instrucciones en dos secuencias ejecutables simultáneamente. Después de llegar al identificador especificado en FORK, se crea un nuevo proceso (secundario) para ejecutar una rama de código bifurcado, mientras que el proceso (principal) de creación continúa ejecutando la otra secuencia. La operación JOIN se utiliza para fusionar las dos secuencias de codificación divididas por FORK y puede existir otro proceso que permita la sincronización con otro proceso secundario.
- **ABORT** (Terminación anormal): Se trata de una terminación forzosa de un proceso, eliminado del sistema a un proceso que funciona mal o sea necesaria su cancelación. El sistema operativo realiza en gran parte las mismas acciones que con la orden DELETE.
- **SUSPEND** (Suspense): Con esta llamada, al proceso designado se la deja indefinidamente en suspenso y se le pone en el estado de suspensión. Sin embargo, sigue en el sistema. Un proceso puede dejarse en suspenso a sí mismo, o a otro proceso cuando está autorizado a hacerlo en virtud de su nivel de privilegio, prioridad o pertenencia.

Un proceso tiene que reanudarse, por lo menos, tantas veces como se le haya dejado en suspenso, para que llegue a estar preparado.

- **RESUME** (Reanudación): Esta llamada reanuda el proceso, que presumiblemente está en suspenso. Obviamente, un proceso en suspenso no se puede reanudar a sí mismo porque tendría que estar en ejecución para conseguir que se procese su llamada al sistema operativo. Por eso, el proceso en suspenso depende de un proceso asociado que emita la llamada RESUME. El sistema operativo responde incluyendo el BCP del proceso en la lista de procesos preparados, con el estado actualizado.
- **DELAY** (Retardo, tiempo de respuesta): Con esta llamada, el proceso destino queda en suspenso durante el período de tiempo especificado. El tiempo se puede expresar en forma de número de señales de reloj del sistema o en unidades de tiempo normales (segundos, minutos). Un proceso se puede retardar a sí mismo, u a otro proceso.

- **GET ATTRIBUTES** (Obtención de atributos): Es una consulta a la que el sistema operativo responde facilitando los valores actuales de los atributos del proceso especificado tomándolo del BCP.

Esta llamada se puede usar para vigilar el estado de un proceso, su utilización de recurso y la información contable u otros datos almacenados en un BCP.

- **CHANGE PRIORITY** (Cambio de prioridad): Se puede usar para modificar la capacidad del sistema por la obtención de recursos. La idea es que la prioridad de un



proceso debe subir y bajar según la importancia relativa de una actividad momentánea, pudiéndose así realizar una planificación más sensible al estado global del sistema.

LLAMADAS DEL SISTEMA⁹	
❖ CONTROL DE PROCESOS:	<ul style="list-style-type: none">➤ FIN, ABORTAR.➤ CARGAR, EJECUTAR.➤ CREAR PROCESO, TERMINAR PROCESO.➤ OBTENER ATRIBUTOS DE PROCESO, ESTABLECER ATRIBUTOS DE PROCESO.➤ ESPERAR UN LAPSO DE TIEMPO.➤ ESPERAR UN SUCESO, INDICAR LA OCURRENCIA DEL SUCESO.➤ ASIGNAR Y LIBERAR MEMORIA.
❖ MANEJO DE ARCHIVOS:	<ul style="list-style-type: none">➤ CREAR ARCHIVO, ELIMINAR ARCHIVO.➤ ABRIR, CERRAR.➤ LEER, ESCRIBIR, REPOSICIONAR.➤ OBTENER ATRIBUTOS DE ARCHIVOS, ESTABLECER ATRIBUTOS DE ARCHIVOS.
❖ MANEJO DE DISPOSITIVOS:	<ul style="list-style-type: none">➤ SOLICITAR DISPOSITIVO, LIBERAR DISPOSITIVO.➤ LEER, ESCRIBIR, REPOSICIONAR.➤ OBTENER ATRIBUTOS DE DISPOSITIVOS, ESTABLECER ATRIBUTOS DE DISPOSITIVOS.➤ CONECTAR Y DESCONECTAR LÓGICAMENTE DISPOSITIVOS.
❖ MANTENIMIENTO DE INFORMACIÓN:	<ul style="list-style-type: none">➤ OBTENER HORA Y FECHA, ESTABLECER HORA Y FECHA.➤ OBTENER DATOS DEL SISTEMA, ESTABLECER DATOS DEL SISTEMA.➤ OBTENER ATRIBUTOS DE UN PROCESO, ARCHIVO O DISPOSITIVO.➤ ESTABLECER ATRIBUTOS DE UN PROCESO, ARCHIVO O DISPOSITIVO.
❖ COMUNICACIONES:	<ul style="list-style-type: none">➤ CREAR, ELIMINAR CONEXIÓN DE COMUNICACIÓN.➤ ENVIAR, RECIBIR MENSAJES.➤ TRANSFERIR INFORMACIÓN DE ESTADO.➤ CONECTAR O DESCONECTAR DISPOSITIVOS REMOTOS.

⁹ Silberschatz – Galván (1999); *Sistemas Operativos: Quinta Edición*; Addison-Wesley Longman de México S.A. de C. V.; página 60.

