

TEMA 6 **BLOQUEOS**

6.1 INTRODUCCION Y EJEMPLOS DE BLOQUEO (O INTERBLOQUEO): Un proceso dentro de un sistema de multiprogramación está *bloqueado* si está esperando por un evento determinado que no ocurrirá. Cuando los recursos son compartidos entre usuarios se debe considerar la *prevención, habitación, detección y recuperación* del interbloqueo y la *postergación indefinida*. (un proceso aunque no esté interbloqueado espera por un evento que probablemente no ocurrirá). A veces el precio de liberar el interbloqueo en sistemas es muy alto y permitirlo es catastrófico.

Los sistemas de cómputos hay recursos que solo se pueden utilizar por un proceso a la vez (impreso, unidad de cinta). El S.O. tiene la capacidad de otorgar temporalmente a un proceso el acceso exclusivo a un recurso o a varios.

Otro Ejemplo es el interbloqueo de un recurso simple en que “el bloqueo se produce en una espera circular de recursos que detiene completamente el tráfico y es necesario una intervención externa para poner orden y reestablecer la normalidad.

Inanición: envejecimiento de un proceso que no usa el procesador. Una solución es que el S.O. eleva la prioridad mientras espera por un recurso, evitando así la postergación indefinida del proceso, sin estar bloqueado o utilizando FIFO.

6.2 CONCEPTOS DE RECURSOS: el S.O. es un administrador de recursos. Los pueden ser “apropiativos” (Ej. CPU, memoria principal), y “no apropiativos” (unidades de cinta).

La apropiatividad es indispensable para el éxito de los sistemas computacionales multiprogramados. Algunos recursos pueden ser compartidos entre varios procesos o dedicados a procesos individuales.

6.3 BLOQUEOS Y CONDICIONES NECESARIAS PARA EL BLOQUEO: si un proceso necesita utilizar un recurso: lo solicita, después lo utiliza y por último lo libera. Si el recurso no está disponible cuando lo solicita debe esperar. En algunos S.O. el proceso se bloquea automáticamente y se despierta cuando dicho recurso está disponible. En otros S.O. la solicitud falla y el proceso debe esperar para luego intentar nuevamente.

Definición formal de bloqueo: un conjunto de procesos se bloquea si cada proceso del conjunto espera un evento que solo puede ser provocado por otro proceso del conjunto. Como todos los procesos están esperando ninguno realizará un evento que pueda despertar a los demás miembros del conjunto. Todos los procesos esperarán por siempre.

Las condiciones necesarias para el bloqueo son (coffman):

Exclusión mutua: el proceso reclama control exclusivo del recurso.

Espera por: los procesos mantienen los recursos que le fueron asignados mientras esperan por recursos adicionales.

No Apropiatividad: los recursos no pueden ser extraídos de los procesos que los tienen hasta su completo uso.

Espera circular: cadena circular de procesos en la que cada uno mantiene uno o más recursos que son requeridos por el siguiente proceso de la cadena.

6.4 MODELACION DE BLOQUEOS: se muestran mediante gráficos dirigidos que tienen dos tipos de modos:

- ◆ Procesos (aparecen como círculos).
- ◆ Recursos (aparecen como cuadrados).
- ◆ $R \rightarrow P$ “poseído”
- ◆ $P \rightarrow R$ “en espera”

■ Las *estrategias utilizadas para enfrentar los bloqueos* son:

- ◆ Ignorar todo el problema.
- ◆ Detección y recuperación.
- ◆ Evitarlos mediante una cuidadosa asignación de recursos.
- ◆ Prevenir mediante la negación de una de las condiciones necesarias.

6.5 AREAS PRINCIPALES EN LA INVESTIGACION DE BLOQUEOS: Principales aspectos:

- *Prevención del bloqueo:* se condiciona un sistema para que se elimine toda posibilidad de que se produzcan. Los resultados dan una pobre utilización de los recursos:
- *Evitación del bloqueo:* impone condiciones menos estrictas que en la prevención para lograr utilizar mejor de los recursos. Si el bloqueo aparece este se lo esquivo.

- **Detección del bloqueo:** Se utiliza en sistemas que permiten que estos ocurran, voluntariamente o no. Determinar si ha ocurrido un bloqueo: se debe detectar con precisión los procesos y recursos implicados, y se lo puede ELIMINAR.
- **Recuperación del bloqueo:** despeja bloqueos de un sistema para continuar operando sin ellos; terminen los procesos estancados y se liberen los recursos. Se logra “extrayendo” (cancelando) procesos bloqueados.

6.6 EL ALGORITMO DEL AVESTRUZ O DE OSTRICH: Se ignora el problema. Algunos s. O. Soportan bloqueos que ni siquiera se detectan, porque que se rompen automáticamente o porque ni siquiera se detectan porque se rompe automáticamente o porque los usuarios prefieren un bloqueo ocasional, en vez de una regla que restringiera a todos los usuarios en el uso de los distintos tipos de recursos. El problema es que se paga un cierto precio para encarar el bloqueo en restricciones para los procesos y en el uso de recursos.

6.7 DETECCION DE BLOQUEOS: El s. O no intenta evitar los bloqueos, intenta detectar cuando han ocurrido y acciona para recuperarse después del hecho.

La detección del bloqueo consiste en determinar si existe o no un bloqueo e identificar que procesos y recursos implicados en el bloqueo.

Los algoritmos de detección implican una sobrecarga en tiempo de ejecución.

Detección de bloqueos de forma:

➔ “Un recurso de cada tipo”: No hay más de un objeto de cada clase de recurso, si la gráfica contiene uno o mas ciclos existe un bloqueo; y cualquier proceso que forme parte del ciclo está bloqueado.

➔ **Varios recursos de cada tipo:** se basan en matrices para detectar un bloqueo entre “N” procesos y “M” recursos de cada clase.

E: es el vector de recursos existentes.

A: es el vector de recursos disponibles (no asignados).

MATRIZ C: asignación actual.

MATRIZ R: solicitudes

Se comparan los vectores con las matrices, estarás bloqueado si lo que se solicita es mayor que lo disponible.

Cuando buscar los bloqueos:

- Cada vez que se solicita un recurso (podría sobrecargar el sistema).
- Verificar cada k minutos.
- Verificar cuando el uso de la cpu baje de cierto valor fijo:

6.8 RECUPERACION DE BLOQUEOS: Para romper el bloqueo de un sistema hay que anular una o mas de las condiciones necesarias para el bloqueo. Normalmente, varios procesos perderán algo de lo realizado hasta el momento.

Factores que dificultan la recuperación de bloqueo:

- ♣ Puede no estar claro si el sistema está o no bloqueado.
- ♣ Muchos sistemas tienen limitaciones para suspender un proceso y luego reanudarlo
- ♣ Los procedimientos de suspensión/reanudación implican sobrecarga.

La recuperación se realiza

- ♣ Retirando forzosamente a un proceso
- ♣ Reclamando sus recursos
- ♣ Permitir que los procesos restantes finalicen

Formas de recuperación ante bloqueos son:

Recuperación mediante la apropiación. Tomar un recurso temporalmente de un proceso y dárselo a otro proceso para que termine.

Recuperación mediante rollback. Se verifica periódicamente los procesos para ver si ocurrió un bloqueo, se graba su estado en un archivo para que más tarde se pueda mirar.

Recuperación mediante la eliminación de procesos. Es la forma mas sencilla de suspender un bloqueo, se elimina un proceso del ciclo o se elimina un proceso que no esta en el ciclo para liberar recursos que necesite un proceso del ciclo. Es preferible eliminar un proceso de compilación que se puede repetir sin problema, que un proceso de actualización de una base de datos.

6.9 EVASION DE BLOQUEOS: se supone que si un proceso solicita un recurso, lo hace solicitando todos al mismo tiempo, pero en la mayoría se solicita uno a la vez. El S.O. decide si otorgar un recurso es seguro o no, y asignarlo si es seguro.

El OBJETIVO es evitar el bloqueo eligiendo correctamente todo el tiempo, pero si necesita información de antemano.

Un ESTADO ES SEGURO si no está bloqueado y existe una forma de satisfacer todas las solicitudes pendientes.

Un algoritmo que intenta evitar los bloqueos es:

- **El algoritmo del banquero (de dijkstra).** Los clientes son los procesos, los créditos son los recursos y el banquero es el S.O.

Consiste en estudiar cada solicitud ver su otorgamiento conduce a un estado seguro y otorga la solicitud, de lo contrario lo pospone. Los procesos deben establecer sus necesidades totales de recursos antes de su ejecución (desventaja) (no utilizado en S.O. de reales)

6.10 PREVENCIÓN DE BLOQUEOS: los bloqueos son imposibles si se garantiza que el menos una de las cuatro condiciones necesarias se satisface (enunciado de Revender).

Estrategias para evitar las condiciones de bloqueo:

- ♣ Cada proceso pide todos sus recursos requeridos de una sola vez.
- ♣ Si un proceso que mantiene sus recursos, se le niega una petición libera los recursos originales y si fuera necesario pedirlos de nuevo junto con los recursos adicionales.

Algoritmos de prevención de bloqueos:

Prevención de la condición de exclusión mutua. Si ningún recurso se asignara de forma exclusiva a un solo proceso, no habrá bloqueos. Imposible de aplicar porque hay recursos no compatibles (Ej. la impresora).

Prevención de la condición “detenerse y esperar” o “espera por”. Se evita que procesos que conservan recursos esperen mas recursos, no hay bloqueos. Se exige a todos solicitar sus recursos de antemano, pero algunos no saben.

Prevención de la condición de “no apropiación”. Los recursos pueden ser retirados de los procesos que los retienen antes de la terminación de los procesos, pero se puede perder información.

Prevención de la condición de “espera circular”. Un proceso solo utiliza un recurso en cada momento, si necesita otro recurso debe liberar el primero; los recursos son enumerados globalmente, y se solicita según el orden (creciente).

6.11 OTROS ASPECTOS:

Cerradura de dos fases. (sistema de bases de dato) consiste en:

- ◆ Primera fase: el proceso intenta cerrar todos los registros necesarios, uno a la vez.
- ◆ Segunda fase: se actualizan y se liberan las cerraduras.

Si durante la primera fase se necesita algun registro ya cerrado, libera todas las cerraduras y comienza en la primera nuevamente. No es aplicable en la realidad, podría haber actualizado, enviando mensajes, etc.

Bloqueos sin recursos. Puede ocurrir que dos procesos se bloqueen en espera de que el otro realice cierta acción.

Inanición. En sistemas dinámicos permanentemente hay solicitudes de recursos, por lo que se necesitará aun criterio para decidir quién obtiene cual recurso y en que momento ya que esto podría ocasionar que ciertos procesos nunca logran el servicio, aun son estar bloqueados, porque su privilegio en el uso de recurso a otro proceso. Se puede solucionar mediante el criterio de asignación de recursos FIFO.

6.12 TENDENCIAS DEL TRATAMIENTO DEL BLOQUEO: el bloqueo tendrá una consideración mayor en los S.O. porque:

- ◆ Orientación hacia la operación asincrónica en paralelo:
- ◆ Ignorancia a priori de los procesos de sus necesidades de recursos.
- ◆ Considerar los datos como un recurso. Incrementar la capacidad del S.O. para administrar mas recursos.