



Tratamiento de interbloqueo

Existen cuatro estrategias para el tratamiento del interbloqueo:

- Ignorar

No tener en cuenta en el sistema operativo este tipo de situaciones, teniendo en cuenta que su aparición es bastante improbable. Utilizada por la mayoría de los sistemas operativos, entre ellos UNIX. La solución es volver a arrancar el sistema.

- Prevenir

Evitar alguna de las cuatro condiciones que deben estar presentes para que pueda aparecer el interbloqueo. Restringen la forma de hacer solicitudes de recursos.

Exclusión mutua: Existen ciertos recursos que pueden ser compartibles como los archivos de solo lectura. En estos casos no requerir exclusión mutua.

Retener y esperar: Un protocolo que puede usarse obliga a cada proceso a solicitar y recibir todos sus recursos antes de iniciar su ejecución.
Otro protocolo permite a un proceso solicitar recursos sólo cuando no tiene ninguno.

No-apropiación: Aplicar el siguiente protocolo: si un proceso que tiene algunos recursos solicita otro que no se le puede asignar de inmediato, entonces todos los recursos que actualmente tiene se liberan implícitamente. Los recursos expropiados se añadirán a la lista de recursos que el proceso está esperando, y este último sólo se reiniciará cuando pueda recuperar sus antiguos recursos y los nuevos que está solicitando.

Espera circular: Imponer un ordenamiento total de todos los tipos de recursos, y exigir que cada proceso solicite recursos en orden creciente de enumeración. Podemos exigir además que un proceso solicite un ejemplar del tipo de recursos R_i , haya liberado cualesquier recursos R_j que tenga, tales que $F(R) \geq F(R_j)$

- Evitar

Se pueden evitar siempre y cuando tengamos disponible cierta información sobre los recursos que necesita cada proceso por adelantado. Existen diversos métodos, entre los que podemos citar el algoritmo del banquero.

Un algoritmo de evitación de bloques examina dinámicamente el estado de asignación de recursos para asegurar que nunca pueda haber una condición de espera circular.

Algoritmo del grafo de asignación de recursos.

Introducimos un nuevo tipo de arista, llamada arista de reserva. Indica que el proceso podría solicitar el recurso en algún instante futuro. Cuando el proceso solicita el recurso esta arista se convierte en una arista de asignación. Cuando el proceso lo libera vuelve a ser de reserva. Los recursos se reservan antes de que el proceso comience su ejecución. Entonces si un proceso solicita un recurso este será asignado si no forma un ciclo.

Algoritmo del banquero



Cada proceso declara el número máximo de ejemplares de cada tipo de recursos que podría necesitar. Hay que mantener varias estructuras de datos para implementar el algoritmo del banquero.

- Detectar y recuperar

Consiste en abortar un proceso cuando existen indicios de que se pueda producir interbloqueo. Para abortar un proceso, el sistema operativo se basa en varias características del mismo:

- Prioridad. Se elimina el menos prioritario.
- Tiempo de procesador usado. Cuando menos tiempo haya estado el proceso a abortar menos trabajo se pierde y más fácil será recuperarlo.
- Tipo de recursos utilizados. Si los recursos son críticos y escasos, será preferible librarlos cuanto antes.
- Necesidades de recursos. Es conveniente examinar aquellos procesos que necesitan un gran número de recursos.
- Facilidad de suspensión/reanudación. Se eliminarán antes aquellos procesos cuyo trabajo perdido sea fácil de recuperar.

Requiere un gasto extra que incluye no solo los costos en tiempo de ejecución de mantener la información necesaria y ejecutar el algoritmo de detección, sino también las pérdidas potenciales inherentes a la recuperación.

Un solo ejemplar de cada tipo de recursos.

Utiliza una variante del grafo de asignación de recursos, llamada grafo de espera. Se obtiene eliminando los nodos de tipo de recursos y uniendo las aristas apropiadas. Una arista de P_i a P_j en un grafo de espera implica que el proceso P_i está esperando que el proceso P_j libere un recurso que P_i necesita. Existe bloqueo si y solo si el grafo de espera contiene un ciclo. Para detectar los bloqueos mutuos, el sistema necesita mantener el grafo de espera e invocar periódicamente un algoritmo que busque ciclos en el grafo.

Varios ejemplares de un tipo de recursos.

Utiliza varias estructuras de datos que varían con el tiempo, similares a las empleadas en el algoritmo del banquero.

Investiga todas las posibles secuencias de asignación para los procesos que todavía no se llevan a cabo.

Recuperación después de un bloqueo mutuo.

Hay dos opciones para romper un bloqueo mutuo:

Terminación de procesos.

Existen dos métodos, en los cuales se recupera el total de los recursos asociados a los procesos finalizados:

- Abortar todos los procesos bloqueados: el costo será elevado pues podría ser que estos procesos hayan trabajado durante largo tiempo, y los resultados de esas operaciones parciales tendrán que desecharse, y tal vez recalcularse después.
- Abortar un proceso a la vez hasta eliminar el ciclo de bloqueo mutuo: incurre en un gasto extra, ya que después de abortar cada proceso se debe invocar un algoritmo de detección de bloqueos mutuos para determinar si todavía hay procesos en bloqueo mutuo.

Expropiación de recursos.

Expropiamos sucesivamente algunos recursos de los procesos y se los damos a otros hasta romper el



ciclo de bloqueo mutuo.

Considerar tres aspectos:

- 1- Selección de la víctima: debemos determinar el orden de expropiación para minimizar el costo.
- 2- Retroceso: debemos regresar el proceso al que le retiramos el recurso a un estado seguro, y reiniciarlo desde ese estado. La solución más fácil es un retroceso total.
- 3- Inanición: debemos asegurar que un proceso dado sólo será escogido como víctima un número finito (pequeño) de veces. La solución más común es incluir el número de retrocesos en el factor de costo.

- Estrategia combinada para el manejo de bloqueos mutuos.

Combina la prevención, la evitación y la detección. El método se basa en la idea de que los recursos pueden dividirse en clases que tienen un ordenamiento jerárquico. Dentro de cada clase se usa la técnica más adecuada para manejar bloqueos mutuos.

No puede haber bloqueos mutuos en el sistema que implementa esta estrategia.