



DESPACHADOR DE PROCESOS

El Sistema Despachador de Procesos es un sistema de multiprogramación con un despachador de procesos de cuatro niveles de prioridad que opera dentro de las limitaciones de los recursos disponibles finitos.

El Despachador de procesos de cuatro niveles:

1. **Procesos de Tiempo Real** que deben ejecutarse inmediatamente basándose en First Come First Served (FCFS), desalojando cualquier proceso que se esté ejecutando con menor prioridad. Estos procesos se ejecutan hasta terminar.
2. **Procesos de Usuario** que son ejecutados mediante un despachador retroalimentado de tres niveles. El quantum del tiempo para cada cola es de 3 segundos para la cola de prioridad 1, 2 segundos para la cola de prioridad 2, y 1 segundo para la cola de prioridad 3.

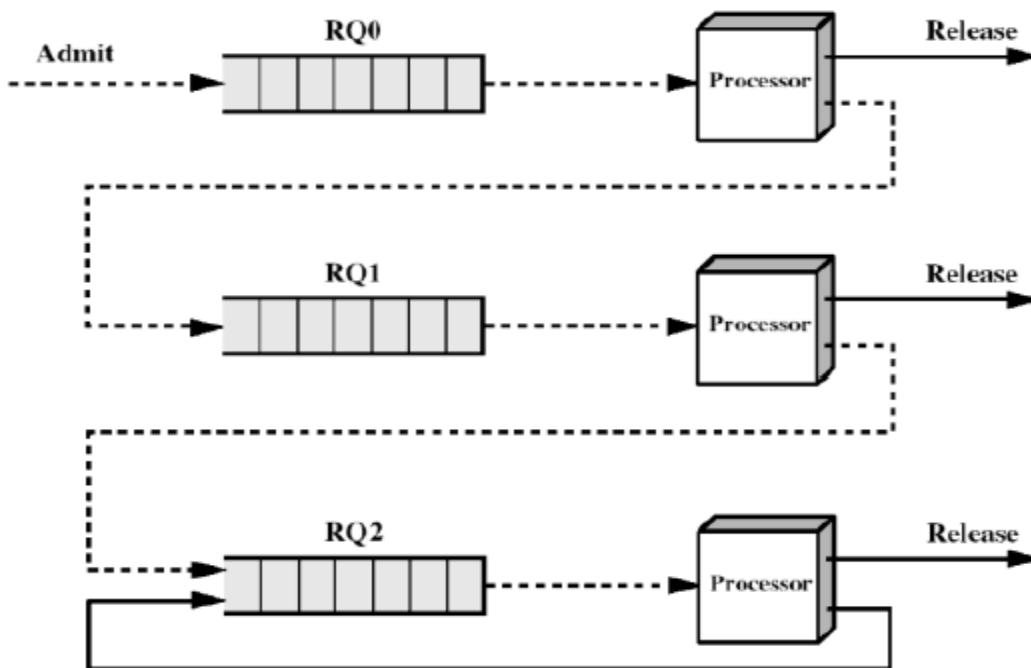


Figura 1. Despachador Retroalimentado de Tres Niveles



El despachador necesita mantener 2 colas(de Tiempo Real y de Usuario) las cuales son alimentadas por la lista despachadora de procesos. Esta lista es examinada en cada instante y los procesos que hayan llegado son luego transferidos a la respectiva cola. Estas colas luego son examinadas, cualquier proceso de Tiempo Real es ejecutado hasta que finalice, desalojando cualquier proceso que esté ejecutándose.

La cola de procesos en Tiempo Real debe estar vacía antes de que se reactive el despachador de las colas retroalimentadas. Cualquier Proceso de Usuario que se encuentre en la cola de Procesos de Usuario y que puedan ejecutarse con los recursos disponibles (dispositivos de E/S) son transferidos a su respectiva cola de prioridades. Normalmente una cola retroalimentada aceptaría todos los trabajos en el nivel más alto y luego degradaría su prioridad cada vez que pasase su quantum de tiempo. Sin embargo, este despachador tiene la habilidad de aceptar trabajos en una prioridad mas baja, insertándolos la cola apropiada para cada proceso. Esto permite al despachador emular un típico Round Robin si todos los procesos aceptados tienen la menor prioridad.

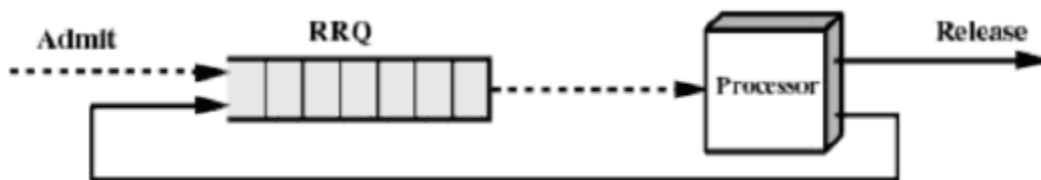


Figura 2. Despachador Round Robin

Cuando todos los procesos de prioridad alta hayan terminado, el despachador retroalimentado se reanuda comenzando o reanudando los procesos que se encuentren al inicio de la cola de mayor prioridad y que no esté vacía. En el siguiente instante el proceso actual es suspendido (o terminado y sus recursos liberados) si hay otro proceso listo de igual o mayor prioridad.



El flujo lógico debería ser igual al mostrado a continuación:

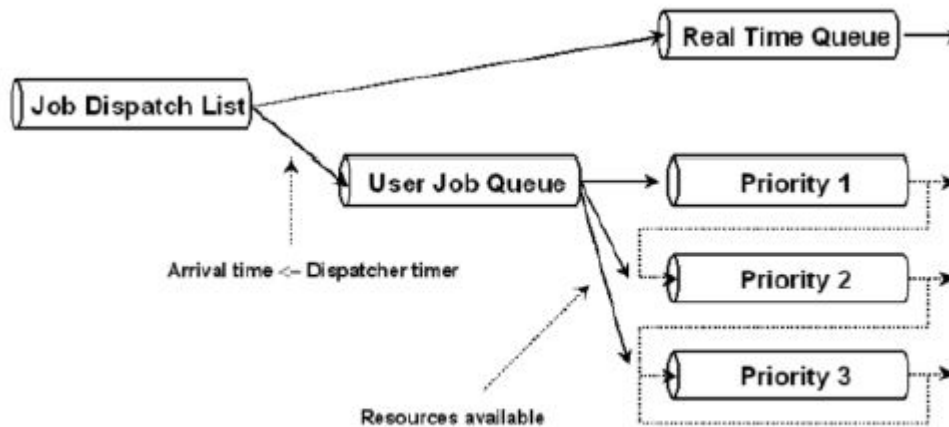


Figura 3. Flujo Lógico

RESTRICCIONES DE RECURSOS

Se poseen los siguientes recursos:

- **2 Impresoras multifuncionales.**
- **3 Modem**
- **2 Lectores de DVD/Blu-ray**
- **1 Cámara web**
- **2 Cornetas**

Los procesos de menor prioridad pueden usar cualquiera o todos los recursos, pero el despachador es notificado sobre cuales recursos el proceso usará. El despachador se asegura de que cada recurso solicitado esta disponible solamente para ese proceso en todo su período de vida en las colas del despachador: desde la transferencia inicial a la cola de procesos hasta las respectivas colas de prioridades y finalizando con su correcta ejecución. Los procesos de Tiempo Real no necesitarán ningún recurso de E/S. Si existen varios procesos esperando por recursos para comenzar su ejecución, se priorizará a los procesos de mayor prioridad para la siguiente asignación de recursos.



PROCESOS

Los procesos son simulados por el despachador creando un nuevo proceso por cada proceso despachado. Cada proceso debe mostrar lo siguiente:

1. Un mensaje que muestre el ID del proceso cuando este empiece.
2. Un mensaje cuando el proceso termine.

El código fuente del proceso hijo se encuentra en el archivo `child.c` y puede ser modificado a su conveniencia.

El ciclo de vida de un proceso es el siguiente:

1. El proceso se envía a las colas de entrada del despachador mediante una lista de procesos inicial que designa el tiempo de llegada (Siempre de menor a mayor), la prioridad (De 0 a 3), el tiempo del procesador requerido (en segundos), el tamaño del bloque de memoria y otros recursos solicitados.
2. Un proceso está listo para ejecutarse cuando sea su turno y todos los recursos necesarios están disponibles para su ejecución.
3. Cualquier trabajo de tiempo real (Prioridad 0) se envía inmediatamente para su ejecución.
4. Si hay suficientes recursos para un proceso de usuario de menor prioridad, el proceso se transfiere a la cola de prioridad adecuada dentro de la unidad de despacho de retroalimentación y se actualizan los indicadores de recursos restantes (dispositivos de E / S)
5. Cuando se inicia un trabajo (**fork()** y **exec()**), el despachador mostrará los parámetros del trabajo (ID del proceso, prioridad, tiempo restante del procesador (en segundos) y recursos solicitados) antes de realizar el **exec**.
6. Se permite ejecutar un proceso en tiempo real hasta que haya expirado su tiempo o cuando el despachador lo mate enviándole una señal SIGINT.
7. Se permite que un trabajo de usuario de baja prioridad se ejecute un segundo del despachador antes de que se suspenda (**SIGSTOP**) o finalice (**SIGINT**) si ha expirado el tiempo. Si se suspende, su nivel de prioridad se reduce (si es posible) y se vuelve a poner en la cola de prioridad apropiada como se muestra en las Figuras 1 y 3 anteriores. El trabajo del Usuario no debe suspenderse y moverse a un nivel de prioridad menor a menos que otro proceso esté esperando ser iniciado.



8. Si no hay trabajos en tiempo real de mayor prioridad pendientes en la cola FCFS, se inicia o reinicia el proceso de mayor prioridad pendiente en las colas retroalimentadas (**SIGCONT**).
9. Cuando finaliza un proceso, los recursos que utiliza se devuelven al despachador para su reasignación a otros procesos.
10. Cuando no hay más procesos en cualquiera de las colas, el despachador finaliza.

LISTA DE DESPACHO

La lista de despacho es una lista de procesos que serán procesados por el despachador. La lista está contenida en un archivo de texto que va a estar definido en la línea de comandos. Por ejemplo:

➤ `despachador listaprocesos.in`

Donde:

-**despachador** sería el ejecutable.

-**listaprocesos.in** sería el archivo con cada uno de los procesos.

Cada línea de la lista describe un proceso con los siguientes datos delimitados por una coma-espacio:

<tiempo de llegada>, <prioridad>, <tiempo de procesador>, <#impresoras>, <#modems>, <#DVD/Blu-ray>, <#camara-web>, <#cornetas>

Ejemplo:

1, 0, 1, 0, 0, 0, 0, 0

1, 1, 2, 1, 0, 1, 1, 2

3, 3, 6, 1, 2, 0, 0, 1

Esto va a significar lo siguiente:

- **1er Proceso:** Tiempo de Llegada: 1, prioridad 0 (Tiempo Real) y necesita 1 seg de CPU– No necesita de E/S.
- **2do Proceso:** Tiempo de Llegada: 1, prioridad 1 (Proceso de Usuario de Alta Prioridad), necesita 2 seg de CPU, 1 impresora, 1 unidad de DVD/Blu-ray, 1 cámara web y 2 cornetas.



- **3er Proceso:** Tiempo de llegada: 3, prioridad 3 (Proceso de Usuario de Baja Prioridad), necesita 6 seg de CPU, 1 impresora, 2 modems y 1 corneta.

El archivo puede ser de cualquier tamaño, es decir que pudiesen haber hasta 1000 procesos en el.

REQUERIMIENTOS DEL PROYECTO

1. Implemente el despachador utilizando el lenguaje C
2. El código fuente **debe** ser extensamente comentado y estructurado apropiadamente. Un código correctamente comentado y presentado es mucho más fácil de interpretar y de corregir.
3. La entrega del código fuente solamente puede contener archivos de extensión **.c** , **.h** y un archivo makefile. **No se debe incluir ningún programa ejecutable.** El makefile reconstruirá automáticamente su programa a partir del código fuente proporcionado, incluyendo el proceso hijo (child.c). Si el código enviado no se compila, no se corrige.
4. La entrega debe estar comprimida en un archivo (tar o zip) y que incluirá todo el código fuente y el makefile (o un shell script de ser necesario).
5. Este proyecto es **estrictamente individual**
6. No se aceptarán entregas fuera de la fecha y hora establecida.
7. De acuerdo con lo establecido en la Ley de Universidades, las copias serán severamente penalizadas colocando cero como calificación a todas las copias encontradas.