

# Despachador

---

- Propósito
- Ejecución
- Operación
- Estructura de datos soporte
- Relación con otros componentes del núcleo

# Propósito

---

- Su propósito: intercambiar procesos entre los procesadores

# Se activa cuando...

---

- (a) Un proceso actual no puede continuar su ejecución, o
- (b) Puede ser que es mejor correr un proceso diferente

... inmediatamente después de que:

---

- 1) Una interrupción externa altera el estado de un proceso
- 2) Una llamada al sistema provoca que el proceso actual no pueda continuar
- 3) Un *trap* causa que el proceso se suspenda, para poder atender la excepción
- *Se ejecuta una rutina de servicio de interrupción.*

# Su operación

---

1. Es el proceso actual el más adecuado?

S: regresa el control al proceso (el HW de interrupción grabó su IP)

3. Graba el descriptor de HW del proceso actual.

# Descriptor de Hardware del proceso

---

- La información del proceso que debe ser guardada antes de interrumpir al proceso
  - Cualquier recurso modificable usado por el proceso
    - Ejemplo: Registros de CPU: AI, RPG, RM, AP
- Formato totalmente dependiente del HW: las instrucciones privilegiadas para guardar/restaurar esperan este formato



# Despachador

---

- Propósito
- Ejecución
- Operación
- Estructura de datos soporte
- Relación con otros componentes del núcleo

# Su operación

---

1. Es el proceso actual el más adecuado?

S: regresa el control al proceso (el HW de interrupción grabó su IP)

3. Graba el descriptor de HW del proceso actual.

4. Restaura el descriptor de HW del proceso más adecuado para correr, obteniéndolo de su PCB.

5. Transfiere el control al nuevo proceso seleccionado, en la localidad indicada por el IP restaurado





# ¿Cuál es el proceso “optimo” para ejecutar?

---

- El de más **prioridad**
- Las prioridades las asignará el **calendarizador** (scheduler)
- Por ahora damos por hecho que están precalculadas

# Cola de procesos

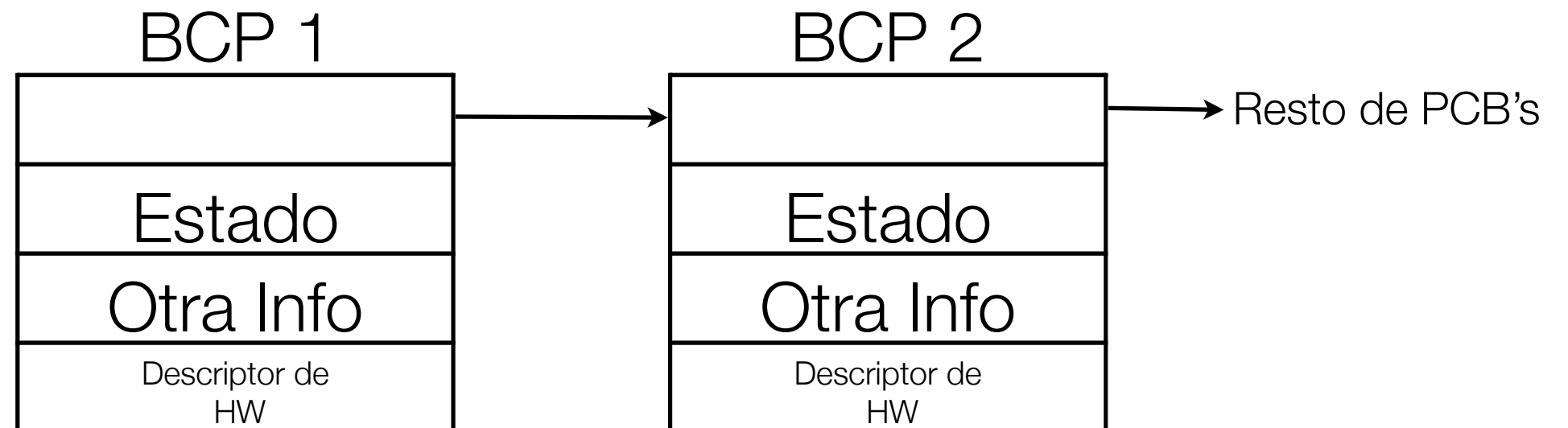
---

- Por simplicidad, es una lista ligada (que extiende a la [estructura de procesos](#)); en donde los procesos [ejecutables](#) aparecen en orden de prioridad decreciente

# Estructura de procesos

---

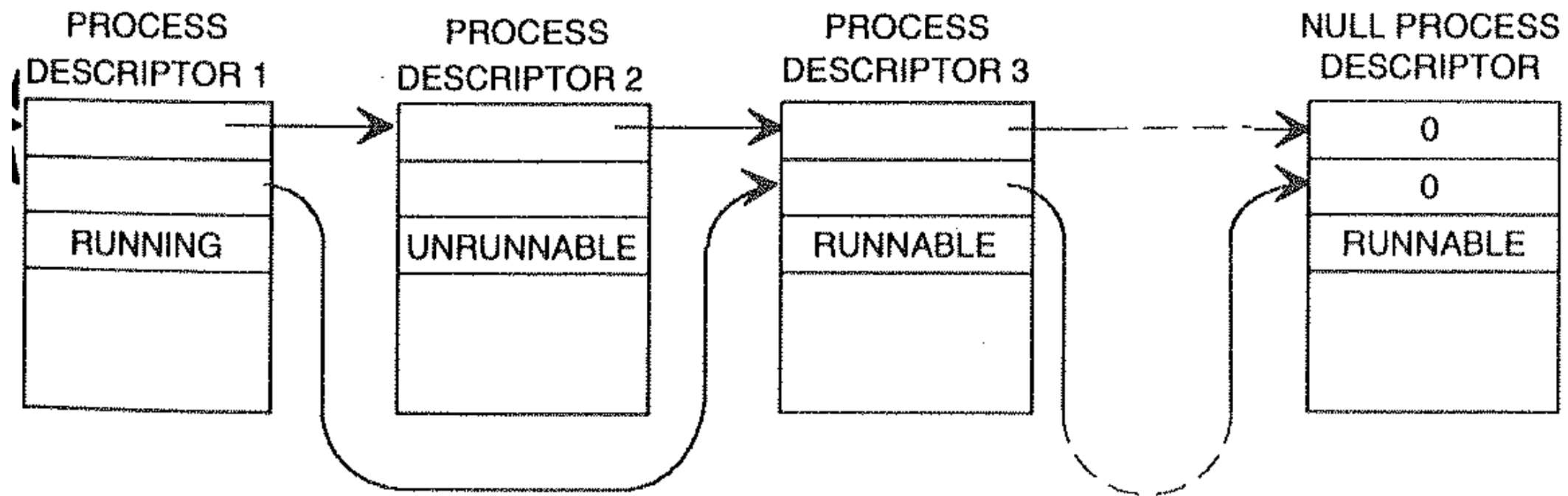
- Provee acceso a los BCP



# Cola de procesos

---

- Por simplicidad, es una lista ligada (que extiende a la [estructura de procesos](#)); en donde los procesos [ejecutables](#) aparecen en orden de prioridad decreciente

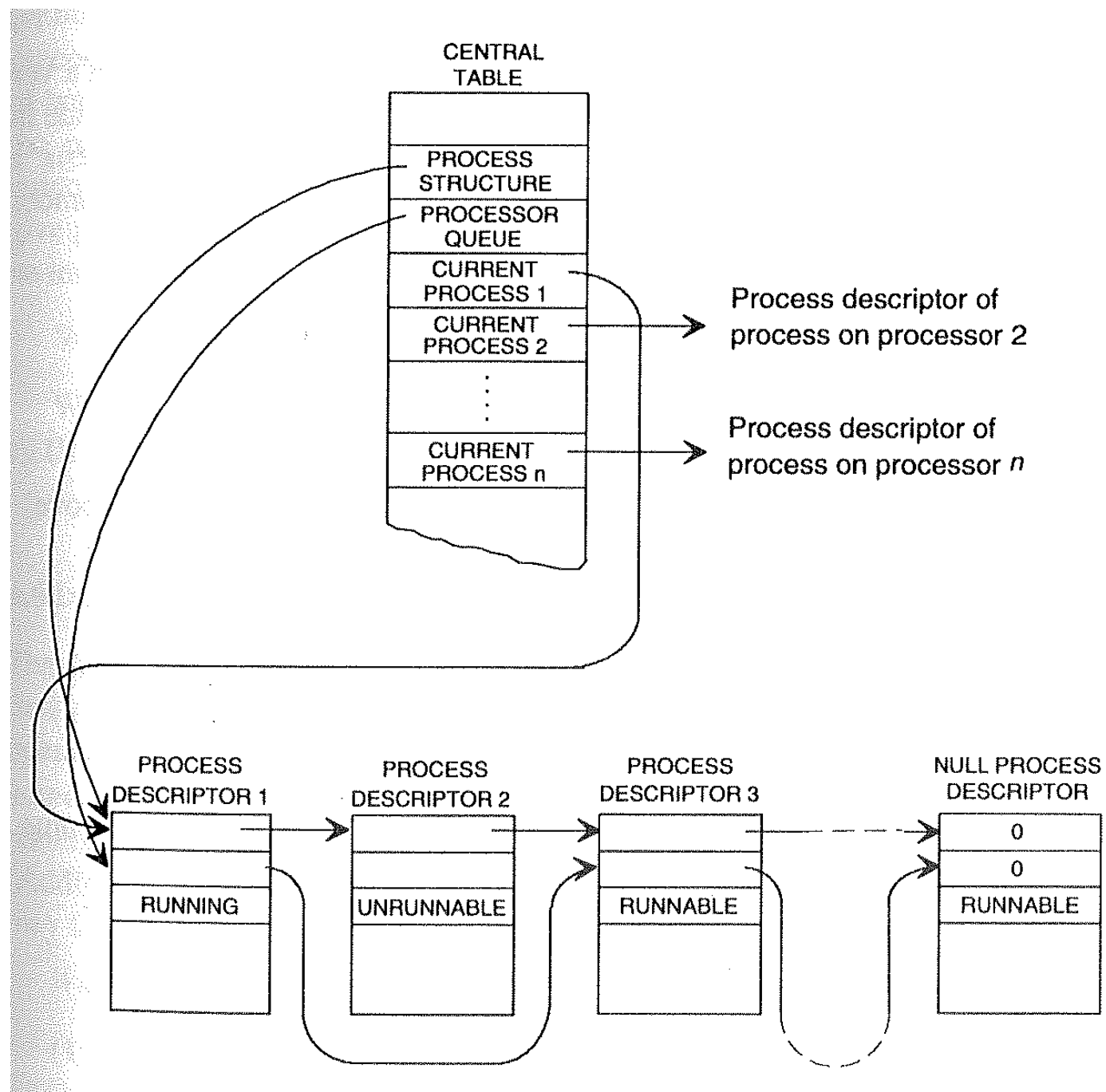


# Cómo se accede a la cola de procesos?

---

# Cómo se accede a la cola de procesos?

- Mediante la tabla central:



# Las rutinas de interrupción, para poder meter un proceso a la cola de procesos deben:

---

- Alterar el estado del proceso en su PCB
- Insertar el proceso en la posición adecuada de la cola de procesos
  - Ésto implica manipular datos “globales”, de forma potencialmente concurrente con otro código que también los manipulan.
  - Se puede hacer con el uso de un semáforo (lo explicaremos después)

# Proceso nulo

---

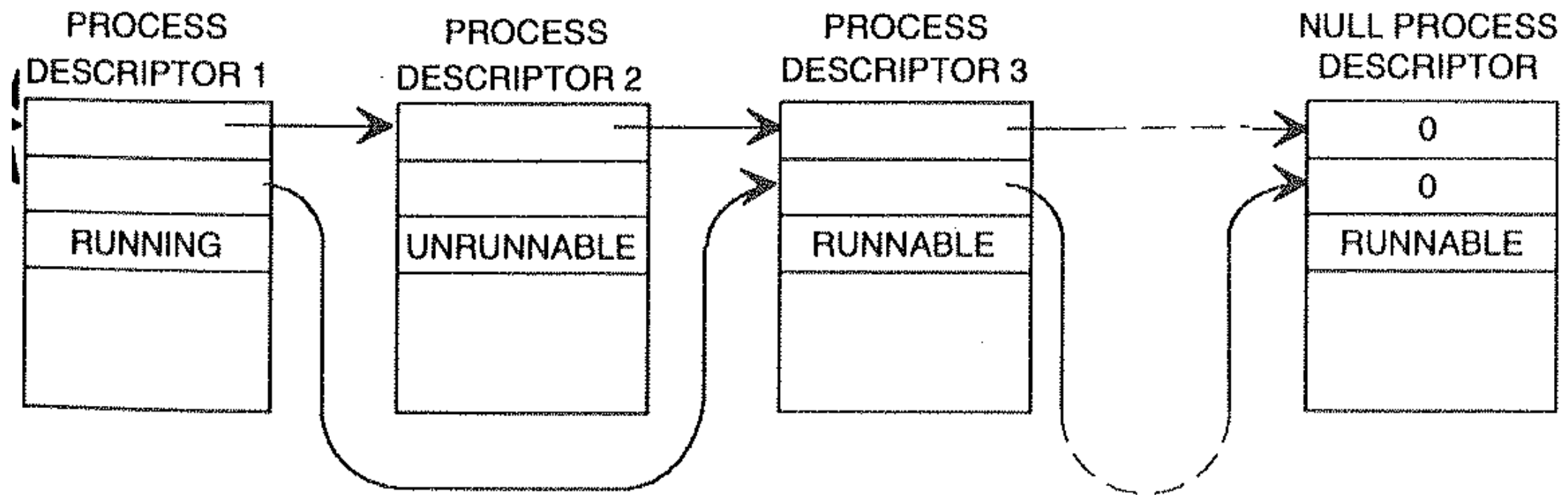
- Qué pasa si hay **menos** procesos ejecutables que procesadores?
  - (Ejemplo Windows)



# Proceso nulo

---

- Qué pasa si hay **menos** procesos ejecutables que procesadores?
- (Ejemplo Windows)



# Ya con la cola del procesador...

---

- Podemos decir que la operación del despachador es la siguiente:
  - Es el proceso actual el proceso al frente en la cola de procesos?

Sí: Continúa su ejecución

- Guarda su descriptor de HW en su PCB
- Restaura el descriptor de HW del proceso al frente de la cola
- Continúa su ejecución

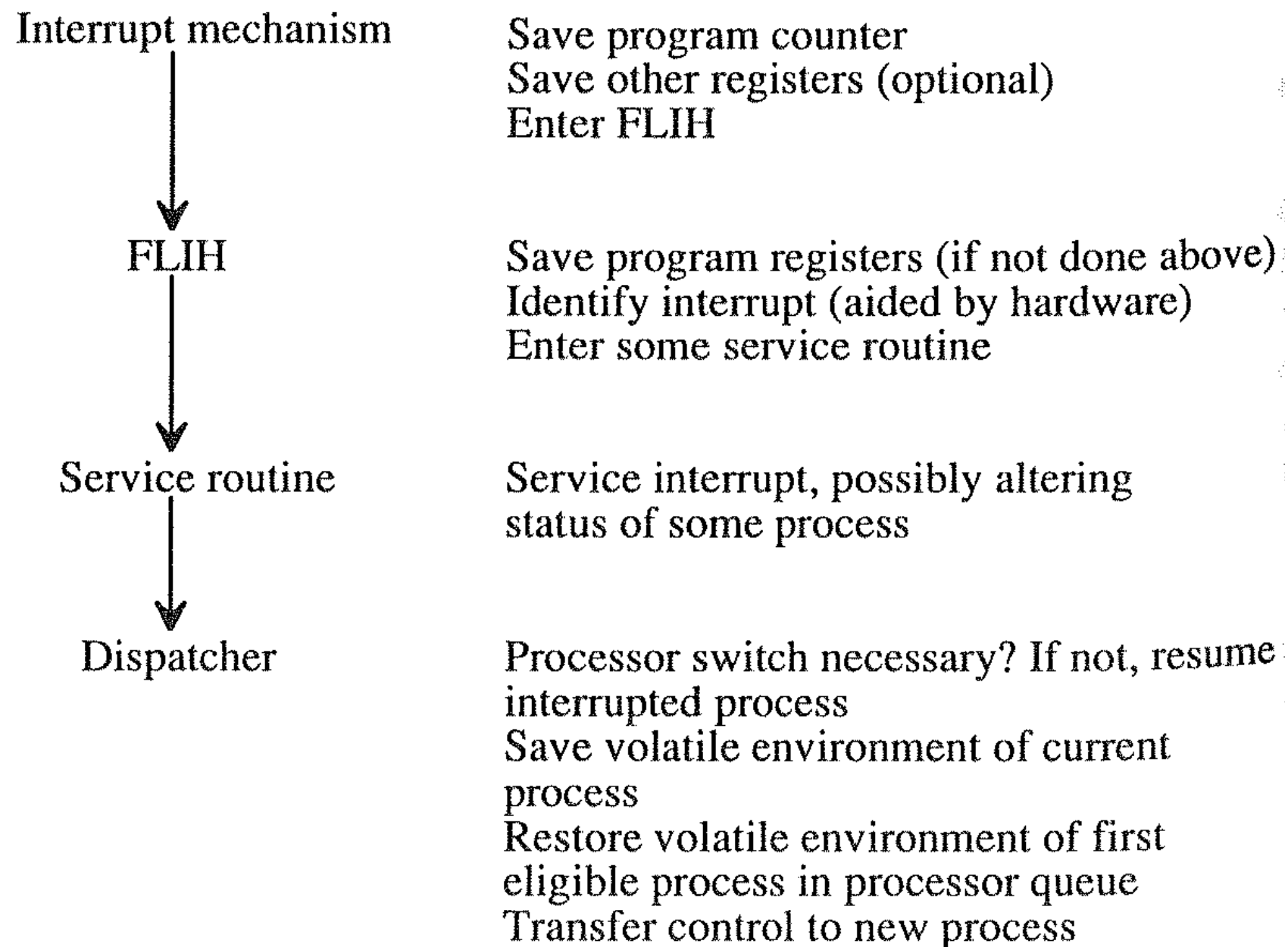
# Cola de procesos, realista?

---

- Las implementaciones reales ocupan un esquema más sofisticado
  - Múltiples colas
  - Demandas de tiempo real
  - ..

# Relación del despachador con el resto de componentes del núcleo

---



# Despachador

---

- Propósito
- Ejecución
- Operación
- Estructura de datos soporte
- Relación con otros componentes del núcleo