

13 ESTRUCTURA DEL ALMACENAMIENTO SECUNDARIO

- 1 ESTRUCTURA DEL DISCO
- 2 ITINERACIÓN DEL DISCO
- 3 ADMINISTRACIÓN DEL DISCO
- 4 ADMINISTRACIÓN DEL ESPACIO DE SWAPPING
- 5 CONFIABILIDAD DEL DISCO
- 6 IMPLEMENTACIÓN DEL ALMACENAMIENTO ESTABLE

13.1 ESTRUCTURA DEL DISCO

- Las unidades de disco son direccionadas como un gran arreglo unidimensional de bloques lógicos, donde el bloque lógico es la unidad más pequeña a transferir.
- El tamaño del bloque lógico suele ser de 512 bytes.

13.1 ESTRUCTURA DEL DISCO

- El arreglo unidimensional de bloques lógicos es mapeado secuencialmente en sectores del disco:
 - El sector 0 es el primer sector de la primera pista (track) en el cilindro más externo.
 - El mapeo procede en orden por esa pista; luego, el resto de las pistas en ese cilindro y después, del más externo al más interno.

13.1 ESTRUCTURA DEL DISCO

- Se debe considerar que:
 - Casi todos los discos tienen sectores defectuosos que se sustituyen con sectores de reserva
 - El número de sectores por pista no es constante, cuánto más lejos del centro, más sectores contiene
- Por lo tanto, los discos modernos se organizan en zonas de cilindros:
 - El número de sectores por pista es constante dentro de la zona
 - A medida que se avanza desde las zonas interiores a las exteriores el número de sectores por pista aumenta
 - Zona exterior = 1.4 más sectores que la zona interior
- Actualmente existen varios miles de cilindros y más de 100 sectores por pista en la zona exterior del disco

13.2 ITINERACIÓN DE DISCO

- El sistema operativo es responsable de que el hardware sea usado eficientemente. Para las unidades de disco, esto quiere decir tener un tiempo de acceso rápido y ancho de banda de disco
- El tiempo de acceso tiene dos componentes principales:
 - **Seek time:** es el tiempo para que el brazo del disco se mueva y se posicione en la pista (o cilindro) que contiene el sector deseado
 - La **latencia rotacional** es el tiempo adicional de espera para que el disco y el sector deseado pase frente a la cabeza lectora/grabadora

13.2 ITINERACIÓN DE DISCO

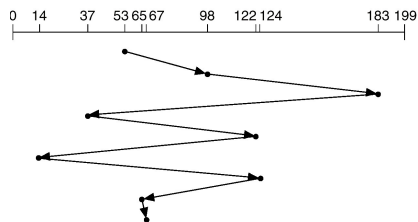
- Minimizar el seek time (tiempo de acceso breve y gran ancho de banda de disco)
- $\text{Seek time} \approx \text{distancia de búsqueda}$
 - El ancho de banda de un disco es el número total de bytes transferidos, dividido por el tiempo total entre el primer requerimiento de servicio y la completación de la última transferencia
- La solididad específica:
 - Si la operación es de entrada o salida
 - La dirección de disco para la transferencia
 - La dirección de memoria para la transferencia
 - El número de bytes por transferir

13.2 ITINERACIÓN DE DISCO

- Existen varios algoritmos para itinerar el servicio de requerimientos de E/S a disco
- A continuación, se ejemplificarán mediante una cola de requerimientos, en un disco de 200 pistas (0-199)
- 98, 183, 37, 122, 14, 124, 65, 67
- La cabeza está posicionada en la pista 53

13.2.1 FCFS (First Come First Served)

El movimiento total de la cabeza L/G es de 640 cilindros



13.2.1 FCFS (First Come First Served)

- Lo ideal sería que las solicitudes de los cilindros 37 y 14 pudieran atenderse juntas, antes o después de las solicitudes para el 122 y el 124.

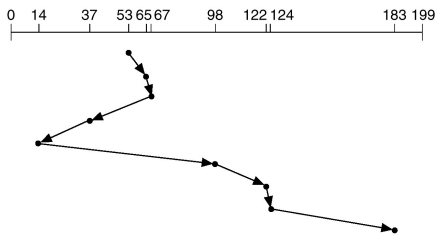
13.2.2 SSTF (Shortest Seek Time First)

- Selecciona el requerimiento con el mínimo seek time desde la posición actual de la cabeza
- SSTF (tiempo de búsqueda más corto) escoge la solicitud pendiente más cercana a la posición actual de la cabeza

13.2.2 SSTF (Shortest Seek Time First)

Secuencia: 53-65-67-37-14-98-122-124-183

- El movimiento total de la cabeza L/G es de 236 cilindros



13.2.2 SSTF (Shortest Seek Time First)

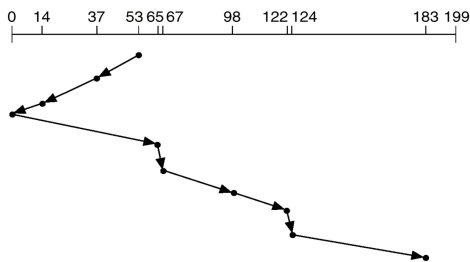
- SSTF es de cierta forma como SJF; puede provocar hambre en algunos requerimientos
- Ej:
 - En cualquier momento podrían llegar más solicitudes, existen dos actualmente: 14 y 186, arriba periódicamente una nueva solicitud cercana a 14. El 186 podría esperar indefinidamente
- ¿es óptimo?

13.2.3 SCAN (barrido)

- El brazo comienza en un extremo del disco y se mueve hacia el otro extremo, atendiendo los requerimientos hasta que llega al otro extremo del disco, donde el movimiento se invierte y el servicio continúa
- A veces es denominado el algoritmo del **ascensor**

13.2.3 SCAN (barrido)

- Moviéndose hacia cero...
- Secuencia: 53-37-14-0-65-67-98-122-124-183
- El movimiento total de la cabeza L/G es de 236 cilindros



13.2.3 SCAN (barrido)

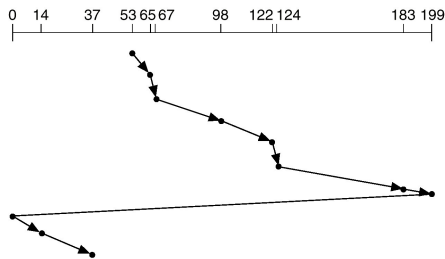
- Si llega una solicitud justo delante de la cabeza, se atenderá casi de inmediato
- Una que quede detrás, tendrá que esperar que llegue al final del disco y cambie de dirección
- Por lo tanto, la mayor cantidad de solicitudes estarán en el otro extremo del disco y son las que esperarán más tiempo.
- ¿Cómo se podría partir por esta parte entonces?

13.2.4 G-SCAN (SCAN Circular)

- Proporciona un tiempo de espera más uniforme que SCAN
- El brazo comienza en un extremo del disco y se mueve hacia el otro extremo, atendiendo los requerimientos hasta que llega al otro extremo del disco, donde "salta" al otro extremo (no atiende requerimientos en el viaje de vuelta) y, nuevamente, comienza a atender los nuevos requerimientos que se produjeron
- Trata a los cilindros como una lista circular que continúa desde el último cilindro hacia el primero

13.2.4 G-SCAN (SCAN Circular)

Secuencia: 53-65-67-98-122-124-183-199 (final) -0 (ppio.) -14-37
movimiento total de la cabeza L/G es de 183+199 (382)
cilindros



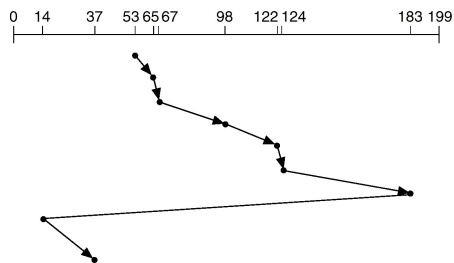
13.2.5 LOOK y G-LOOK

- Versión de G-SCAN: mueve el brazo a todo lo ancho de disco ¿se implementará de esta forma?
- El brazo sólo va lo más lejos posible en cada dirección, según el requerimiento; luego, invierte la dirección inmediatamente, sin ir hasta el final del disco.
- "Mira" si hay una solicitud antes de continuar en esa dirección

13.2.5 LOOK y G-LOOK

Secuencia: 53-65-67-98-122-124-183-14-37

movimiento total de la cabeza L/Ges de 322 cilindros



13.2.6 Selecc. de un Algoritmo de Itinerac.

- SSTF es común y tiene una apariencia más natural
- SCAN y G-SCAN se comportan mejor para sistemas con una carga de trabajo fuerte sobre el disco. No son prodivos al problema de hambre o inanición
- El desempeño depende del número y tipo de solicitudes por ej: si en general sólo hay una solicitud pendiente ¿cómo se comporta?
- El método de asignación de archivos puede influir mucho sobre las solicitudes: contiguo (solicitudes son cercanas), enlazado o por índices (bloques dispersos).

13.2.6 Selecc. de un Algoritmo de Itinerac.

- Ubicación de directorios y bloques de índices también debe considerarse: abrir un archivo que está en un directorio -> abrir el directorio primero (ubicado en el primer cilindro) y luego abrir el archivo (está en el último cilindro)
- El algoritmo de itineración de disco debería ser escrito aparte del Sistema Operativo, permitiendo ser reemplazado con uno diferente si fuera necesario
- SSTF y LOOK son razonables como algoritmo por omisión

13.2.6 Selecc. de un Algoritmo de Itinerac.

- En discos modernos, la latencia rotacional puede ser casi tan grande como el tiempo de búsqueda promedio.
- Es complejo reducir la latencia pues los discos no revelan la ubicación física, sólo la lógica al S.O.
- Los fabricantes de discos, implementan algoritmos de planificación -> HW del controlador incorporado en la unidad del disco.
- El S.O. envía una gran cantidad de solicitudes, el controlador las deja en una cola y los planifica

13.3 ADMINISTRACIÓN DE DISCO

- 1 Formateo de Disco
- 2 Sector de Arranque (Boot)
- 3 Bloques Defectuosos

13.3.1 Formateo de Disco

- Formateo de bajo nivel (físico): divide un disco en sectores que el controlador puede leer y escribir
- Requiere de estructura de datos especiales para el sector:
 - Cabeecera
 - Área de datos (comúnmente de 512 bytes)
 - Apéndice.

13.3.1 Formateo de Disco

- Para usar un disco que almacene archivos, el sistema operativo también necesita registrar sus propias estructuras de datos en el disco
 - Particionar el disco en uno o más grupos de cilindros (se trata como si fuera un disco individual)
 - Formateo lógico o "creación del sistema de archivos" (mapas de espacio libre y asignado, directorio inicial vacío)

13.3.2 Sector de Arranque (Boot)

- El sistema se inicializa (se enciende o reinicia) desde el Sector de arranque:
 - Asigna valores iniciales a todos los aspectos del sistema (registros de CPU, controladores, contenido de la mem. ppal, finalmente pone en marcha al S.O.)
 - El S.O. está en disco y lo carga a mem ppal. "saltando" a una dirección inicial para partir

13.3.2 Sector de Arranque (Boot)

- La rutina de inicialización está cargada en ROM (Read Only Memory) ¿puede infectarse con un virus? ¿cambiar chip o programa cargador?
- Programa de carga de la inicialización se almacena en la partición denominada bloques de arranque

Organización de disco de MS-DOS: bloque de 512 bytes para prog. De arranque

Sector de Arranque
FAT
Directorio Raíz
Bloques de Datos (subdirectorios)

13.3.3 Bloques Defectuosos

- Piezas móviles y la cabeza se mueve cerca de la superficie del disco ¿son propensos a fallas?
- Muchos sectores defectuosos
- Los datos se pierden
- Utilización de métodos tales como sector de repuesto, usado para manejar bloques defectuosos, se sustituyen por defectuosos (cálculo del mismo cilindro)

13.3.3 Bloques Defectuosos

- Se puede sustituir también por deslizamiento de sectores:
 - Bloqué lógico 17 defectuosos
 - Reserva disponible: sector 203
 - Se mueven todas una posición
 - 202 a 203, 201 a 202...18 a 19, 17 a 18.
- En general podría requerirse intervención humana ¿por qué?

13.4 ADMINIST.R. DEL ESPACIO DE SWAP

- La memoria virtual utiliza al disco como una extensión de la memoria física.
- El acceso a disco es más lento -> tiempo de intercambio afecta el rendimiento
- Objetivo del diseño e implementación del espacio de intercambio:
 - Mejorar el rendimiento al sistema de memoria virtual
- Se debe considerar:
 - Uso del espacio de intercambio
 - Colocación en disco
 - Administración

13.4 ADMINISTR. DEL ESPACIO DE SWAP

- Uso del espacio de intercambio
 - Depende de los algoritmos de gestión de memoria
 - Podría contener toda la imagen del proceso (código y datos) o almacenar las páginas que son expulsadas de memoria ppa.
 - Varía:
 - Cantidad de memoria física
 - Cantidad de memoria virtual y la manera en que se utiliza
 - ¿es mejor sobreestimar o subestimar el espacio de intercambio?

13.4 ADMINISTR. DEL ESPACIO DE SWAP

- Ubicación del espacio de intercambio:
 - Sistema de archivos
 - Partición de disco aparte
- Sistema de archivos: fácil de implementar, es ineficiente (navegar la estructura de directorios consume tiempo)
- Partición: más común de usar, administrador de espacio para asignar y liberar bloques
 - Usa algoritmos que favorecen la rapidez en vez de la eficiencia de almacenamiento
 - ¿cuál es la desventaja?
- En general se usa un híbrido

13.4 ADMINISTR. DEL ESPACIO DE SWAP

Administración del espacio de Swapping (ej. De unix)

- El núcleo emplea dos mapas de intercambio:
 - Segmento de texto tiene tamaño fijo

Mapa

512k	512k	512k	71k				...	
------	------	------	-----	--	--	--	-----	--

Mapa de intercambio del segmento de texto en 4.3BSD

- Segmento de datos, puede crecer con el tiempo

Mapa

16k	32k	64k	128k	256k			...	
-----	-----	-----	------	------	--	--	-----	--

Mapa de intercambio del segmento de datos en 4.3BSD

13.4 ADMINISTR. DEL ESPACIO DE SWAP

Administración del espacio de Swapping

- 4.3BSD asigna espacio de Swap cuando un proceso comienza, manteniendo un segmento de texto (el programa) y un segmento de datos
- El Kernel usa mapas de intercambio para hacer un seguimiento del uso del espacio
- Solaris 2 asigna espacio de Swapping sólo cuando una página es forzada a salir de la memoria física, no cuando la página de memoria virtual es creada por 1ª vez

13.5 CONFIABILIDAD DE DISCOS

- Tienen tasas de fallos relativamente altas
- Pérdida de datos
- Tiempo de inactividad importante mientras se reemplaza el disco y se restauran los datos
- Mejorar la confiabilidad es un tema de investigación relevante

13.5 CONFIABILIDAD DE DISCOS

- Se han hecho varias mejoras en las técnicas de uso de disco, las que incluyen el uso de múltiples discos trabajando en forma cooperativa
- Se puede usar un grupo de discos como una unidad de almacenamiento
- El esquema RAID mejora el rendimiento y la confiabilidad del sistema de almacenamiento guardando datos redundantes
 - Mediante Mirroring (espejo) o Shadowing (sombra), se almacena un duplicado de cada disco. Caro.
 - Con paridad entrelazada de bloques, se usa mucho menos redundancia. Rápido, menos costoso y confiable.

13.6 IMPLEM. DE ALMACENAM. ESTABLE

- Un esquema de registro write-ahead (escritura adelantada) requiere de un almacenamiento estable
- Una escritura puede tener:
 - Finalización con éxito
 - Fallo parcial
 - Fallo total
- Se detecta y se invoca un procedimiento de recuperación para restaurar

13.6 IMPLEM. DE ALMACENAM. ESTABLE

- Se mantienen dos bloques físicos por cada bloque lógico:
 - Se escribe la info. en el primer bloque
 - Finaliza con éxito, se escribe en el segundo bloque físico
 - Se declara la operación completa cuando finaliza con éxito la segunda escritura
- Para implementar un almacenamiento estable:
 - Replicar información en más de un medio de almacenamiento no volátil, con modo independiente de falla
 - Actualizar la información en un controlador, de manera de asegurar que se puedan recuperar los datos estables luego de cualquier falla durante la transferencia o recuperación de datos
