Algoritmos e Estruturas de Dados II - SCC-203

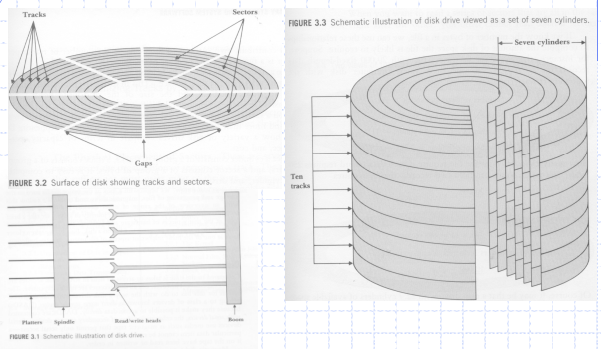
Arquivos: Discos

Gustavo Batista

Organização da informação no disco

- Disco: conjunto de 'pratos' empilhados:
  - Dados são gravados nas superfícies desses pratos.
- Superfícies: são organizadas em trilhas.
- Trilhas: são organizadas em setores.
- Cilindro: conjunto de trilhas na mesma posição.

Organização da informação no disco



Organização da informação no disco

- http://www.youtube.com/watch?v=kmp01dYXQcc

Endereços no disco

- Um setor é a menor porção endereçável do disco.
- Exemplo:
  - fread(&c,1,1, fd): lê 1 byte na posição corrente
    - S.O. determina qual a superfície, trilha e setor em que se encontra esse byte;
    - O conteúdo do setor é carregado para uma memória especial (buffer de E/S) e o byte desejado é lido do buffer para a RAM. Se o setor necessário já está no *buffer*, o acesso ao disco torna-se desnecessário.

Seeking

- Movimento de posicionar a cabeça de L/E sobre a trilha/setor desejado.
- O conteúdo de todo um cilindro pode ser lido com 1 único *seek*.
- É o movimento mais lento da operação leitura/escrita.
- Deve ser reduzido ao mínimo.

Cluster

- Conjunto de setores logicamente contíguos no disco.
- Um arquivo é visto pelo S.O. como um grupo de clusters distribuído no disco:
  - Arquivos são alocados em um ou mais clusters.

FAT – *File Allocation Table*

- Cada entrada na tabela dá a localização física do cluster associado a um certo arquivo lógico.
- 1 *seek* para localizar 1 cluster :
  - Todos os setores do *cluster* são lidos sem necessidade de *seeks* adicionais.
- Reduz o tamanho da FAT.

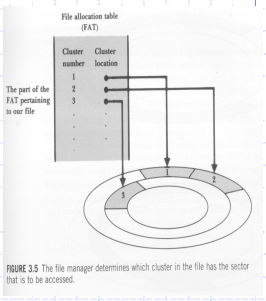


FIGURE 3.5 The file manager determines which cluster in the file has the sector that is to be accessed.

Extent

- Seqüência de *clusters* consecutivos no disco, alocados para o mesmo arquivo.
- 1 *seek* para recuperar 1 extent.
- A situação ideal é um arquivo ocupar 1 extent:
  - freqüentemente isso não é possível, e o arquivo é espalhado em vários *extents* pelo disco.

Extent

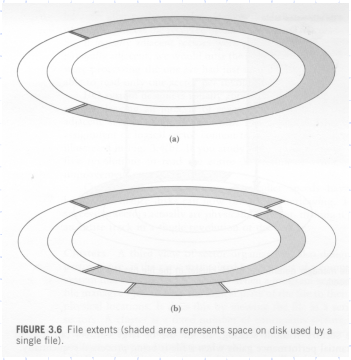


FIGURE 3.6 File extents (shaded area represents space on disk used by a single file).

Capacidade do disco**

- Capacidade do setor:
  - nº bytes (Ex. 512 bytes).
- Capacidade da trilha:
  - nº de setores/trilha * capacidade do setor.
- Capacidade do cilindro:
  - nº de trilhas/cilindro * capacidade da trilha.
- Capacidade do disco:
  - nº de cilindros x capacidade do cilindro.

**Capacidade Nominal

Fragmentação

- Perda de espaço útil decorrente da organização em setores de tamanho fixo.
- Fragmentação também ocorre organizando os arquivos em clusters:
  - Ex: 1 cluster = 3 setores de 512 bytes, arquivo com 1 byte (quanto espaço se perdeu?)

Sistema de Arquivos

- A organização do disco em setores/trilhas/cilindros é uma formatação física (já vem da fábrica).
- É necessária uma formatação lógica, que 'instala' o sistema de arquivos no disco:
  - Subdivide o disco em regiões endereçáveis;
  - Grava estruturas de gerenciamentos dos arquivos.

Sistema de Arquivos

- O sistema de arquivos FAT (DOS/Windows) não endereça setores, mas grupos de setores (clusters):
  - 1 cluster = 1 unidade de alocação;
  - 1 cluster = *n* setores.
- Um arquivo ocupa, no mínimo, 1 cluster:
  - Unidade mínima de alocação.
- Se um programa precisa acessar um dado, cabe ao sistema de arquivos do S.O. determinar em qual cluster ele está (FAT).

Tamanho do *cluster*

- Definido pelo S.O. quando o disco é formatado.
- (FAT DOS/Windows): Determinado pelo máximo que a FAT consegue manipular, e pelo tamanho do disco:
  - FAT16: pode endereçar 2^{16} clusters = 65.536.
- Quanto maior o cluster, maior a fragmentação!

Outros sistemas de arquivos

- FAT32 (Windows 95 e posteriores):
  - clusters de tamanho menor, endereça mais clusters, menos fragmentação.
- NTFS (New Technology File System):
  - Sistemas OS/2 (IBM) e Windows NT;
  - Mais eficiente: a menor unidade de alocação é o próprio setor de 512 bytes.

Custo de acesso a disco

- É uma combinação de 3 fatores:
  - Tempo de busca (seek): tempo p/ posicionar o braço de acesso no cilindro correto;
  - Delay de rotação: tempo p/ o disco rodar de forma que a cabeça de L/E esteja posicionada sobre o setor desejado;
  - Tempo de transferência: tempo p/ transferir os bytes.

Tempo de Busca (Seek)

- É a parte mais expressiva do tempo de acesso.
- Depende de quanto o braço precisa se movimentar.
- É geralmente mais caro em ambientes multi-usuário.
- Para cálculos, se trabalhar com o tempo de busca médio (tempo de busca para 1/3 do número de cilindros).

Delay de Rotação

- Por exemplo, para um HD de 5000 rpm, o *delay* de rotação é de 12ms.
- Na média, considera-se o *delay* de rotação de meia-volta.
- Na prática, esse *delay* é reduzido quando é possível ler/gravar o arquivo em setores da mesma trilha e trilhas do mesmo cilindro.

Tempo de Transferência

- Tempo transferência = (nº de bytes transferidos / nº de bytes por trilha) * tempo de rotação.
- Exemplo: disco de 10000 rpm com 170 setores por trilha:
  - Para ler 1 setor: 1/170 de rotação;
  - 10000 rpm = 6ms por rotação;
  - Tempo de transf. para 1 setor = 0,035ms.

Exemplo de Cálculo de Tempo de Acesso

- Supor duas situações:
  - o arquivo está gravado seqüencialmente em um disco;
  - o arquivo está espalhado pelo disco, cada *cluster* em uma trilha diferente.
- Objetivo:
  - Comparar os tempos de acesso e a influência do tempo de busca (*seek*).

Exemplo de Cálculo de Tempo de Acesso

- Arquivo:
  - 8.704 kbytes;
  - 34.000 registros de 256 bytes.
- HD/S.O.:
  - 1 setor = 512 bytes;
  - 1 *cluster* = 8 setores = 4096 bytes;
  - 1 trilha = 170 setores.
- São necessários:
  - 2125 *cluster* ou
  - 100 trilhas.

Exemplo de Cálculo de Tempo de Acesso

- Primeira situação:
  - O arquivo está disposto em 100 trilhas.
- Supor os seguintes tempos (Seagate Cheetah):
  - Tempo de busca médio: 8 ms;
  - *Delay* de rotação: 3 ms;
  - Tempo de leitura de 1 trilha: 6 ms;
  - Total: 17 ms.
- Tempo total = 100 x 17 ms = 1.7 s.

Exemplo de Cálculo de Tempo de Acesso

- Segunda situação:
  - 2125 *clusters* alocados dispersos pelo disco.
- Supor os seguintes tempos (Seagate Cheetah):
  - Tempo de busca médio: 8 ms;
  - *Delay* de rotação: 3 ms;
  - Tempo de leitura de 1 *cluster*: 0.28 ms;
  - Total: 11.28 ms.
- Tempo total = 2125 x 11.28 ms = 23.9 s.

Disco como gargalo

- Discos são muito mais lentos que as redes locais ou a CPU.
- Muitos processos são "*disk-bound*", i.e, CPU e rede têm que esperar pelos dados do disco.

Técnicas p/ minimizar o problema

Multiprogramação: CPU trabalha em outro processo enquanto aguarda o disco.

Stripping: o arquivo é repartido entre vários drives (paralelismo).

RAID: (*redundat array of inexpensive disks*):

- Stripping (RAID 0), Mirroring (RAID 1);
- http://linas.org/linux/raid.html.

Técnicas p/ minimizar o problema

Disk cache: blocos de memória RAM configurados para conter páginas de dados do disco. Ao ler dados de um arquivo, o *cache* é verificado primeiro. Se a informação desejada não é encontrada, um acesso ao disco é realizado, e o novo conteúdo é carregado no *cache*.

RAM Disk: simula em memória o comportamento do disco mecânico.

Agradecimentos

- Slides da profa. Cristina Ferreira.

28