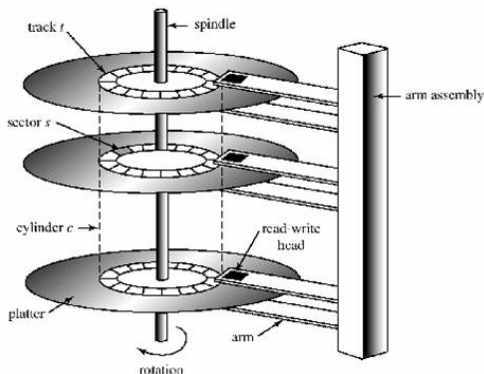


Transferência de Dados

Helton Hideraldo Bísaro

Transferência de Dados

O que ocorre quando um programa escreve um valor em um arquivo em um disco?



Transferência de Dados

O que ocorre quando um programa escreve um valor em um arquivo em um disco?

- 1 O programa pede ao S.O. para escrever o conteúdo de uma variável na próxima posição disponível do arquivo
- 2 O sistema operacional repassa a tarefa para o gerenciador de arquivos (file manager), que é um subconjunto do SO.
- 3 O gerenciador de arquivos busca em uma tabela informações sobre o arquivo, como por exemplo: se está aberto e disponível para uso; tipos de acessos são permitidos; arquivo físico corresponde o arquivo lógico;
- 4 O gerenciador de arquivos busca em uma FAT a localização física do setor (ou do cluster, ou do extent) onde deve ser colocado o byte.

Transferência de Dados

- 5 O gerenciador de arquivos armazena o setor num buffer de I/O do sistema (se ele já não estiver lá); a seguir o valor é armazenado na sua posição correta
- 6 O gerenciador de arquivos instrui o processador de I/O sobre onde o valor está na RAM e onde precisa ser gravado no disco.
- 7 O processador de I/O formata o dado apropriadamente, e decide o melhor momento de escrevê-lo no disco.
- 8 O processador de I/O envia o dado para o controlador de disco.
- 9 O controlador de disco verifica se o drive está disponível (ele pode estar atendendo outro processo), instrui o drive de disco para mover a cabeça de L/E para a trilha correta, espera o setor desejado ficar sob a cabeça e, então, envia o byte para o disk drive que o deposita, bit-a-bit na superfície do disco

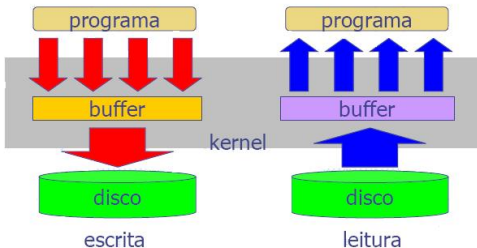
Gerenciamento de Buffer

- Os blocos transferidos de memória secundária vão ser alojados em zonas de memória RAM denominados **buffers** e aí mantidos enquanto for possível.
- O conteúdo de um bloco de disco em buffer é denominado **página**.
- Dado que o espaço de buffer é limitado, é a substituição de blocos em memória (páginas) por blocos de disco. Esta substituição está sujeita a **critérios**.

Gerenciamento de Buffer

Quantos buffers devem ser utilizados?

- Exemplo: imagine um sistema com um único buffer realizando, intercaladamente, operações de Read e Write. **Problemas?**
- Sim, e por isso a maioria dos sistemas utiliza pelo menos 2 buffers, sendo um para leitura e outro para escrita.



Gerenciamento de Buffer

Critérios de substituição de páginas:

- **LRU (Least Recently Used)** - a página correspondente ao bloco menos utilizado é a melhor candidata a ser substituída.
- **MRU (Most Recently Used)** - a página correspondente ao bloco de disco mais utilizado ultimamente é a melhor candidata a ser substituída.
- Quando não é possível prever quais os blocos que irão ser solicitados em seguida \Rightarrow o futuro não será muito diferente do presente (princípio de localidade) \Rightarrow as páginas mais solicitadas não deverão ser substituídas \Rightarrow LRU.
- Segundo este critério as páginas lidas e as páginas escritas devem permanecer no buffer.

Gerenciamento de Buffer

Multiplos buffers - estratégias

Mesmo no caso de transferência em uma única direção a existência de um único buffer retarda consideravelmente as operações.

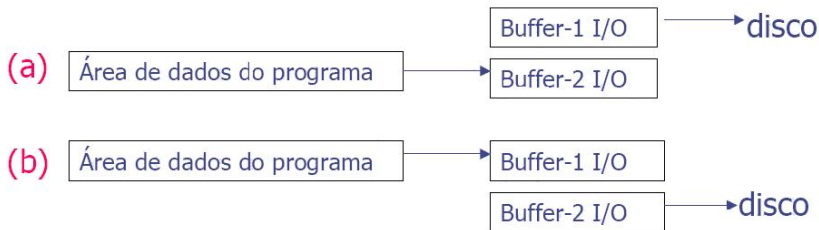
- É demorado preencher um buffer com o conteúdo de todo um setor.
- A solução é usar múltiplos buffers para leitura/escrita

A multipla buferização (Multiple buffering)

- Se vários buffers são utilizados, e o sistema permite sobreposição de tarefas de I/O e CPU, a CPU pode ficar enchendo um buffer enquanto o conteúdo do outro é transmitido para o disco
- O S. O. gerencia os buffer, em geral, o programador não tem acesso a eles.

Gerenciamento de Buffer

Multiplos buffers - estratégias



Gerenciamento de Buffer

Multiplos buffers - estratégias

- Quando chega uma requisição de um setor para o S.O. ele primeiro verifica se ele está em algum buffer
- Se não está ele coloca o setor em algum buffer livre.
- Se não há buffer livre então o S. O. deve:
 - 1 escolher um buffer ocupado e escrever no disco o conteúdo corrente do buffer
 - 2 Há diversas políticas para escolha de um buffer, entre elas a do menos recentemente atualizado
 - 3 E colocar o setor requisitado nesse buffer

Gerenciamento de Buffer

Outras estratégias

MOVE MODE

- não distingue área de dados do programa dos buffers do sistema.
- Evita que os dados sejam movidos da área de dados do programa para os buffers de I/O e vice-versa.

LOCATE MODE

- gerador de arquivos fornece aos programas, ponteiros para os buffers ou
- o gerador de arquivos faz I/O diretamente entre o disco e área de dados do programa

Custos de Acesso a Disco

Alguns conceitos importantes

- **Tempo de seek:** é o tempo gasto para a cabeça de leitura/gravação se posicionar na trilha correta. Varia de 3 ms (para trilhas adjacentes) e até 100 ms (para trilhas que estão nos extremos do disco).
- **Latência rotacional:** é o tempo gasto para localizar o setor ao qual se quer ter acesso.
- **Tempo de transferência:** é o tempo gasto para a migração dos dados da memória secundária para a memória principal.

Custos de Acesso a Disco

Exemplo:

- 1 O que ocorre quando um arquivo é acessado sequencialmente, e o arquivo está armazenado em vários cilindros consecutivos? \Rightarrow o tempo do seek é reduzido.
- 2 O que ocorre quando dois arquivos, localizados em extremos opostos do disco (um no cilindro mais externo, e outro no mais interno), são acessados alternadamente? \Rightarrow o tempo do seek é alto. Isso se torna mais crítico ainda em ambientes multiusuário.

Custos de Acesso a Disco

Tempos base para cálculo - Exemplo:

Tempo min de seek	6 mesc
Tempo médio de seek	18 msec
Latência rotacional	8.3 msec
Taxa máxima de transferência	16.7 msec/trilha ou 1229 bytes/msec
Bytes por setor	512
Tamanho do cluster	8 setores
Menor extent	5 clusters
Trilhas por cilindro	11
Trilhas por superfície	1331
Fator de intercalação	1

Custos de Acesso a Disco

Exercício: Calcular o tempo necessário para ler um arquivo de 2.048 kbytes, dividido em 8.000 registros de 256 bytes cada.

- O espaço é alocado para os arquivos em unidades de uma trilha → extent
- Cada setor armazena 2 registros → Cada cluster pode armazenar 16 registros
- Assim, o arquivo ocupará 500 clusters ($8000/16$).
- Como o menor extent é 5 clusters 100 extents → 100 trilhas.
- Vamos assumir que as 100 trilhas estão espalhadas aleatoriamente no disco (pior caso)

Custos de Acesso a Disco

Exercício:

Calcular o tempo necessário para ler um arquivo de 2.048 kbytes, dividido em 8.000 registros de 256 bytes cada.

Tempo de acesso depende do tipo de acesso:

- Sequencial → o máximo do arquivo é processado a cada acesso;
- Aleatório → apenas um registro é acessado por vez.

Custos de Acesso a Disco

Custo com acesso sequencial

Custo = tempo para leitura do arquivo, setor por setor, em sequência.

- Para cada trilha (sequência de setores consecutivos), o processo de leitura envolve as seguintes medidas:
 - tempo médio de seek: 18 msec
 - latência rotacional: 8.3 msec
 - tempo de transferência : 16.7 msec (leitura de uma trilha)
- Custo de acesso a uma trilha = $18 + 8.3 + 16.7 = 43$ msec
Custo de acesso ao arquivo:
- Para 100 trilhas: $100 \times 43 \text{ msec} = 4.300 \text{ msec} = \mathbf{4.3 \text{ sec}}$

Custos de Acesso a Disco

Custo com acesso aleatório

Custo = Em vez de ler um setor depois do outro, o acesso é feito por registros, o que envolve mudar de trilha a cada vez que um novo registro é lido.

- Para cada registro, essa operação envolve:
 - tempo médio de seek: 18 msec
 - latência rotacional: 8.3 msec
 - tempo de transferência de um cluster: 3.3 msec ($16,7 \times 1/5$)
- Custo de acesso a um registro = $18 + 8.3 + 3.3 = 29.6$ msec
Custo de acesso ao arquivo:
- Para 8.000 registros: $8.000 \times 29.6 \text{ msec} = 236.800 \text{ msec} = 236.8 \text{ sec.} \approx \mathbf{4 \text{ min.}}$

Custos de Acesso a Disco

Discos como gargalo

- Os discos estão ficando cada vez mais rápidos, mas são muito mais lentos que as redes.
- Enquanto drives de disco de alto desempenho possuem taxa de pico de transferência de 3 MB/seg, redes de alto desempenho conseguem transmitir dados a uma taxa de pico de 100MB/seg.
- A taxa de pico de transferência informa quantos MB são transferidos por segundo em uma situação ideal, i.e., é o melhor que o drive consegue fazer.
- isso normalmente significa que um processo é diskbound , ou seja, CPU e rede têm que esperar pelos dados sendo transmitidos pelo disco.

Custos de Acesso a Disco

Discos como gargalo

Técnicas usadas para minimizar o problema.

- Multiprogramação: permite que a CPU faça outras coisas enquanto espera pelos dados.
- striping: divisão do arquivo em partes, alocando cada parte em um disco diferente, e fazendo com que cada disco transmita partes do arquivo simultaneamente (requer múltiplos drives de disco).
- RAM disk: uma porção enorme de RAM configurada para simular o comportamento do disco (normalmente utilizadas para simular de discos flexíveis).
- disk cache: uma porção de memória configurada para conter páginas de dados do disco. Quando o dado é requisitado, o conteúdo do cache é verificado para ver se já não contém a informação desejada.

Custos de Acesso a Disco

Discos como gargalo

Disk Cache.

- O tamanho do buffer cache é limitado, necessitando políticas para substituição de blocos como FIFO ou Last Recently Used (LRU).
- A falta de energia pode acarretar perda de dados que foram modificados no cache e não foram atualizados no disco.
 - Solução 1: Atualizar periodicamente em disco todos os blocos modificados no cache.
 - Solução 2: Atualizar imediatamente no disco toda a vez que os blocos são modificados no cache.