Arquitetura de Computadores

Aula T26– 09 Junho de 2023

Dispositivos de E/S orientados para a transferência de blocos: discos de tecnologia magnética e SSD; RAID

Bibliografia:

OSTEP Caps. 37, 38 e 44

https://pages.cs.wisc.edu/~remzi/OSTEP/file-disks.pdf

https://pages.cs.wisc.edu/~remzi/OSTEP/file-raid.pdf

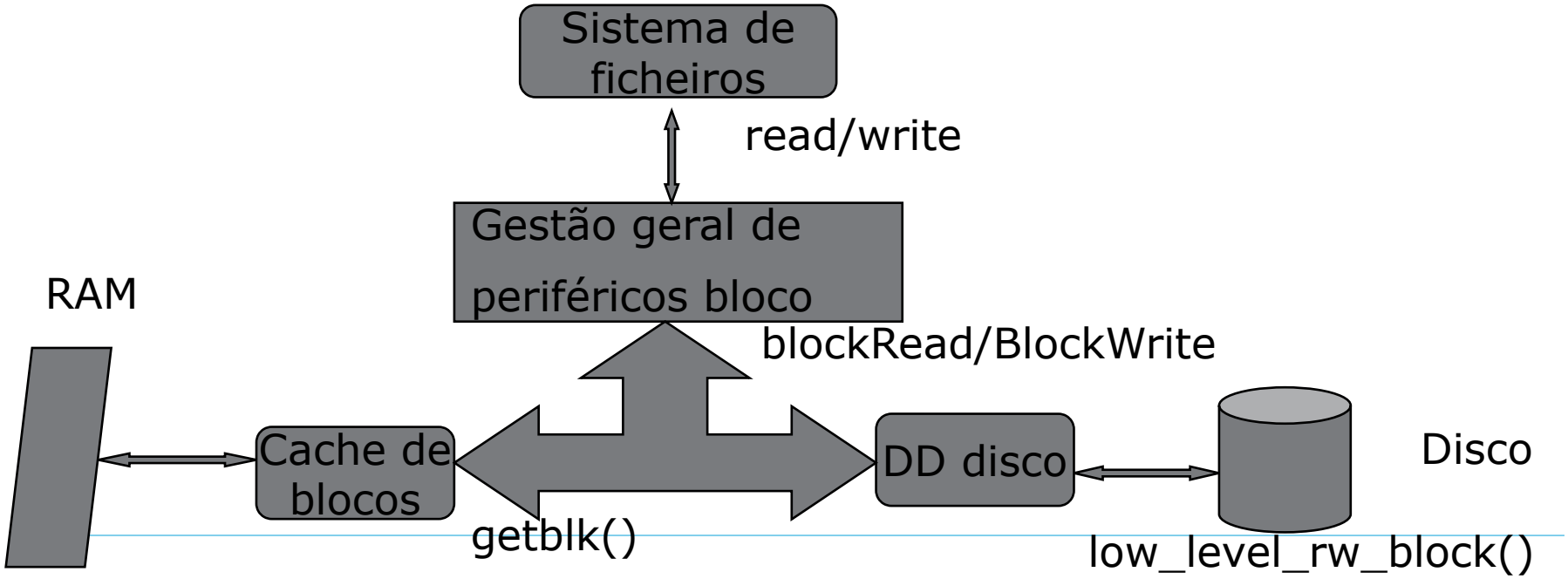
https://pages.cs.wisc.edu/~remzi/OSTEP/file-ssd.pdf

Dispositivos orientados para a transferência de blocos

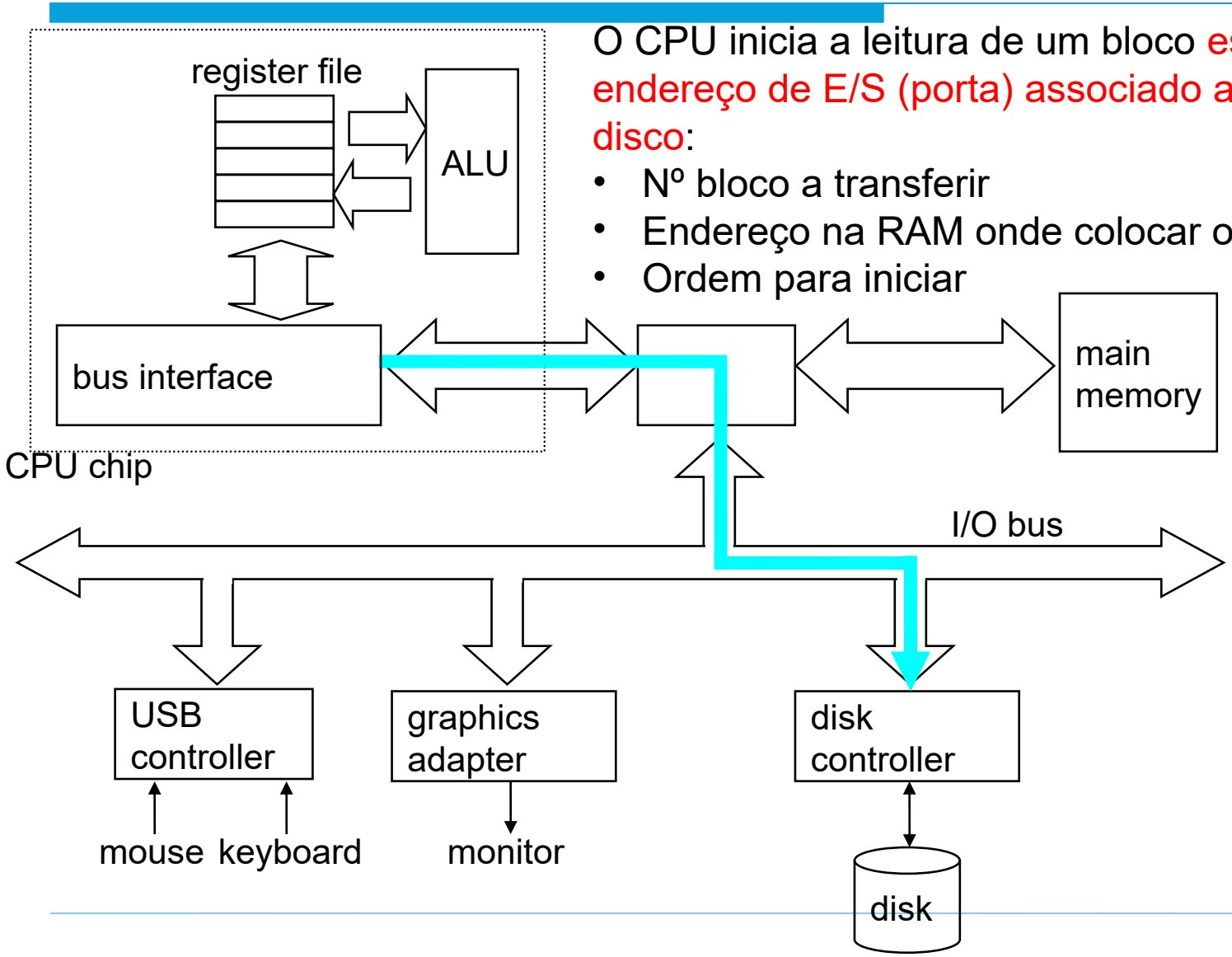
- Dispositivos tipo bloco (ex: discos)
  - Comandos: ler_bloco, escrever_bloco
  - “Raw I/O” – acesso directo aos blocos – normalmente só para programas com privilégios especiais
  - Através do sistema de ficheiros

Periféricos tipo bloco

- Tipicamente discos rígidos – latência elevada, taxa de transferência elevada
- Optimização é ler muitos bytes contíguos numa operação única: cache de blocos de disco



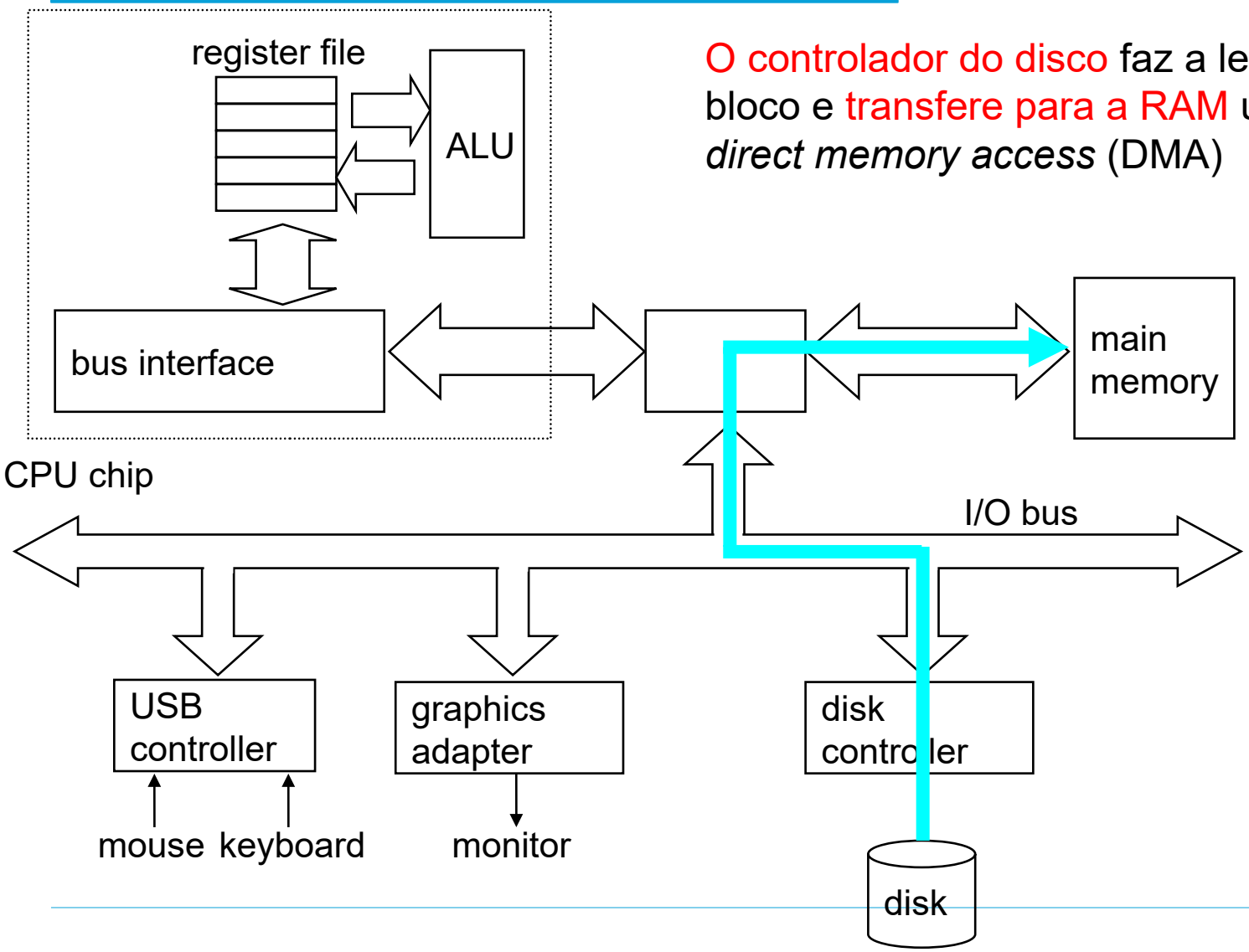
Leitura de um bloco do disco: Passo 1



O CPU inicia a leitura de um bloco escrevendo num endereço de E/S (porta) associado ao controlador do disco:

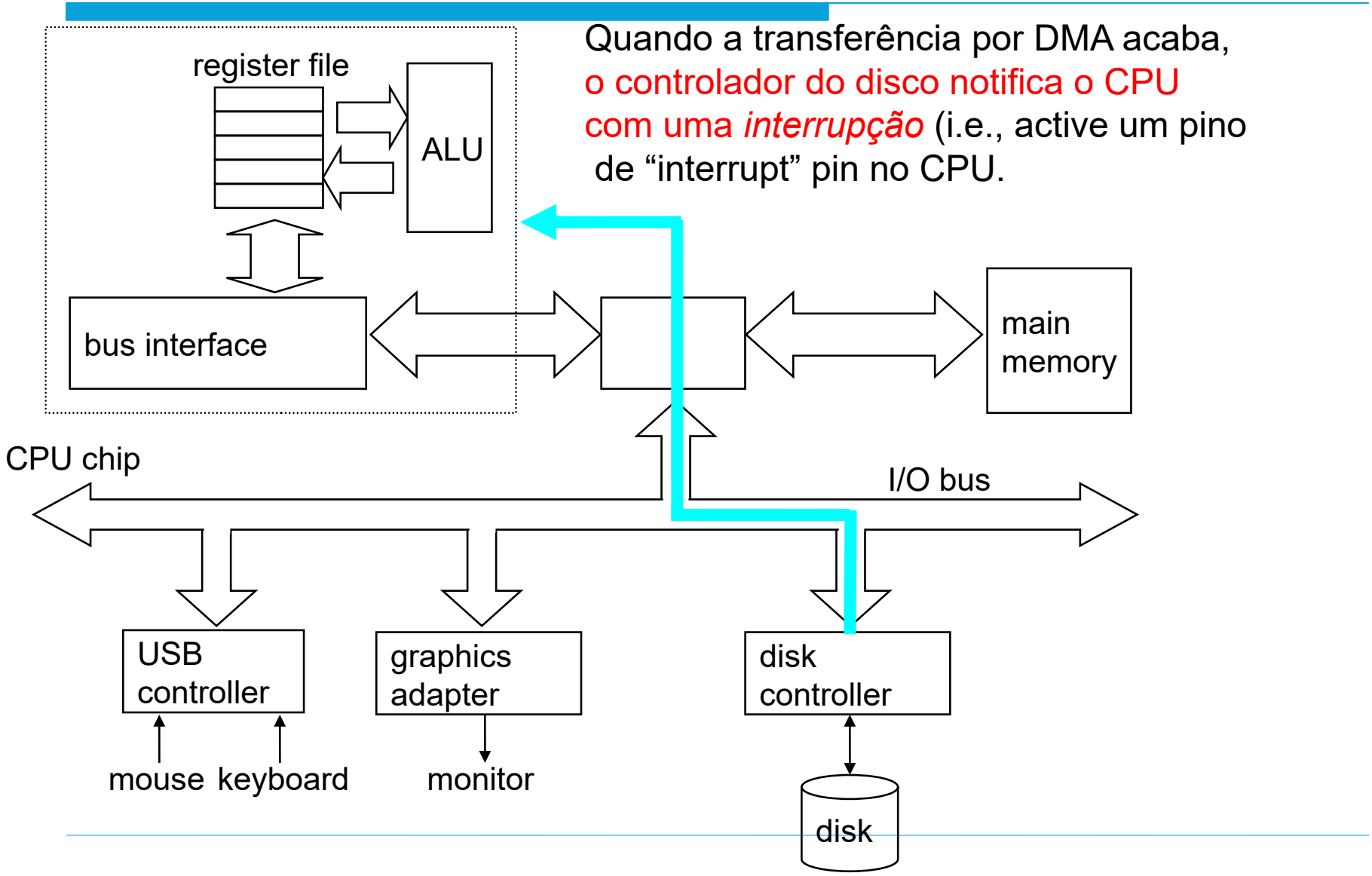
- Nº bloco a transferir
- Endereço na RAM onde colocar o conteúdo
- Ordem para iniciar

Leitura de um bloco do disco: Passo 2



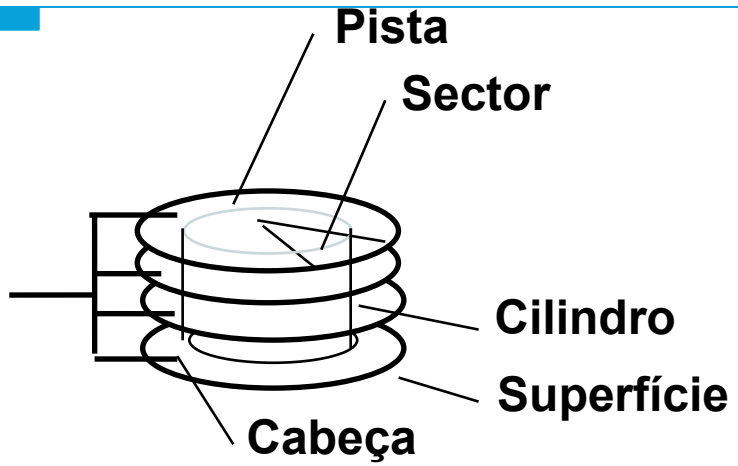
O controlador do disco faz a leitura do bloco e transfere para a RAM usando *direct memory access* (DMA)

Leitura de um bloco do disco: Passo 3



Discos magnéticos

- Objectivo:
  - armazenamento não volátil a longo prazo
  - grande capacidade, nível mais lento na hierarquia de memória
- Características:
  - Seek Time (~8 ms média)
    - latência posicional
    - latência de rotação
- Ritmo de transferência
  - Aprox. 1 sector per ms (5-15 MB/s)
  - Blocos
- Capacidade: Gigabytes e quadruplica de 3 em 3 anos



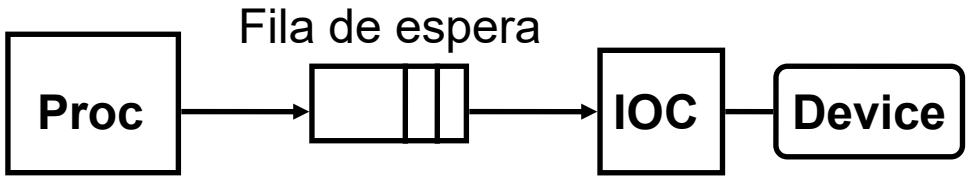
7200 RPM = 120 RPS => 8 ms por rot.
Latência de rotação média = 4 ms
128 sectors por pista => 0.25 ms por sector
1 KB por sector => 16 MB / s

Tempo de resposta
= Fila + Controlador + Seek + Rot + Xfer

Tempo de serviço

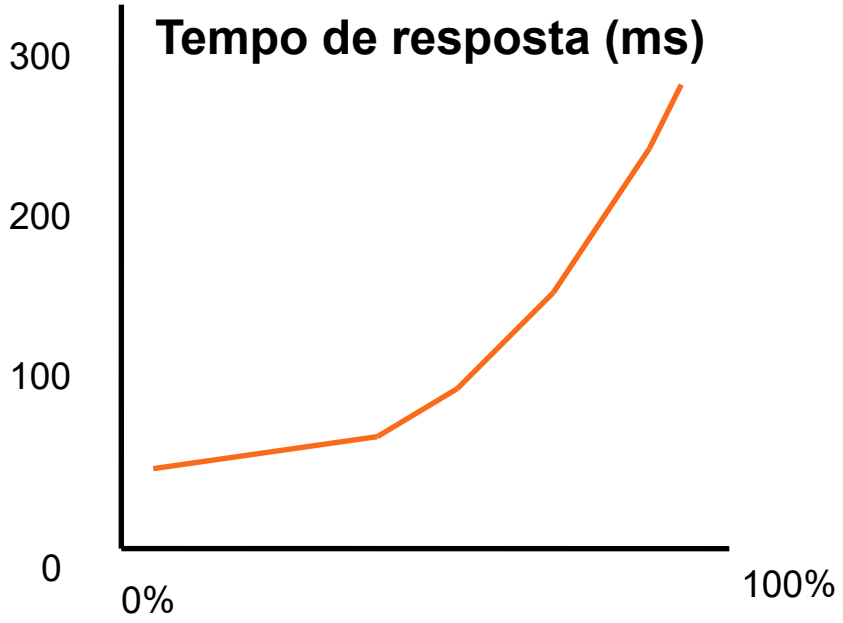
Performance do I/O de disco

Tempo de resposta = Tempo na Fila + Tempo de Serviço do Disco



Teoria das filas de espera
Tr – tempo de resposta
Ts – tempo de serviço
Tq – tempo de espera na fila
U - taxa de utilização
U = nº de pedidos/s * Ts

Tr = Ts/ (1- U)



Tempo de resposta de um disco

- Parâmetros do disco:
  - Tamanho do bloco é 8K bytes
  - Seek time publicitado é de 12 ms
  - Disco roda a 7200 RPM
  - taxa de transferência é 4 MB/sec
- Overhead no controlador de 2 ms
- Suponhamos que o disco está sempre livre - não há tempo de espera
- Qual é o tempo médio de acesso a um sector?
  - T. seek médio + rot delay médio + tempo transf. + ov. controlador
  - 12 ms + 0.5/(7200 RPM/60) + 8 KB/4 MB/s + 2 ms
  - 12 + 4.15 + 2 + 2 = 20 ms
- O seek time assume que não há localidade: o real é tipicamente menor do que o anunciado:

Parâmetros que ditam o tempo de acesso ao disco (2)

- Tempo de acesso
  - Soma do seek time com o rotational delay (exemplo 7ms+4ms)
  - Tempo de transferência desprezável: a transferência faz-se enquanto o sector passa debaixo da cabeça (no exemplo do WD 8ms/280 sectores ~ 28 uS por sector (17 Mbytes/s)
- Naturalmente, o tempo de acesso de acesso a um ficheiro será tanto menor quanto mais os seus sectores estiverem próximos

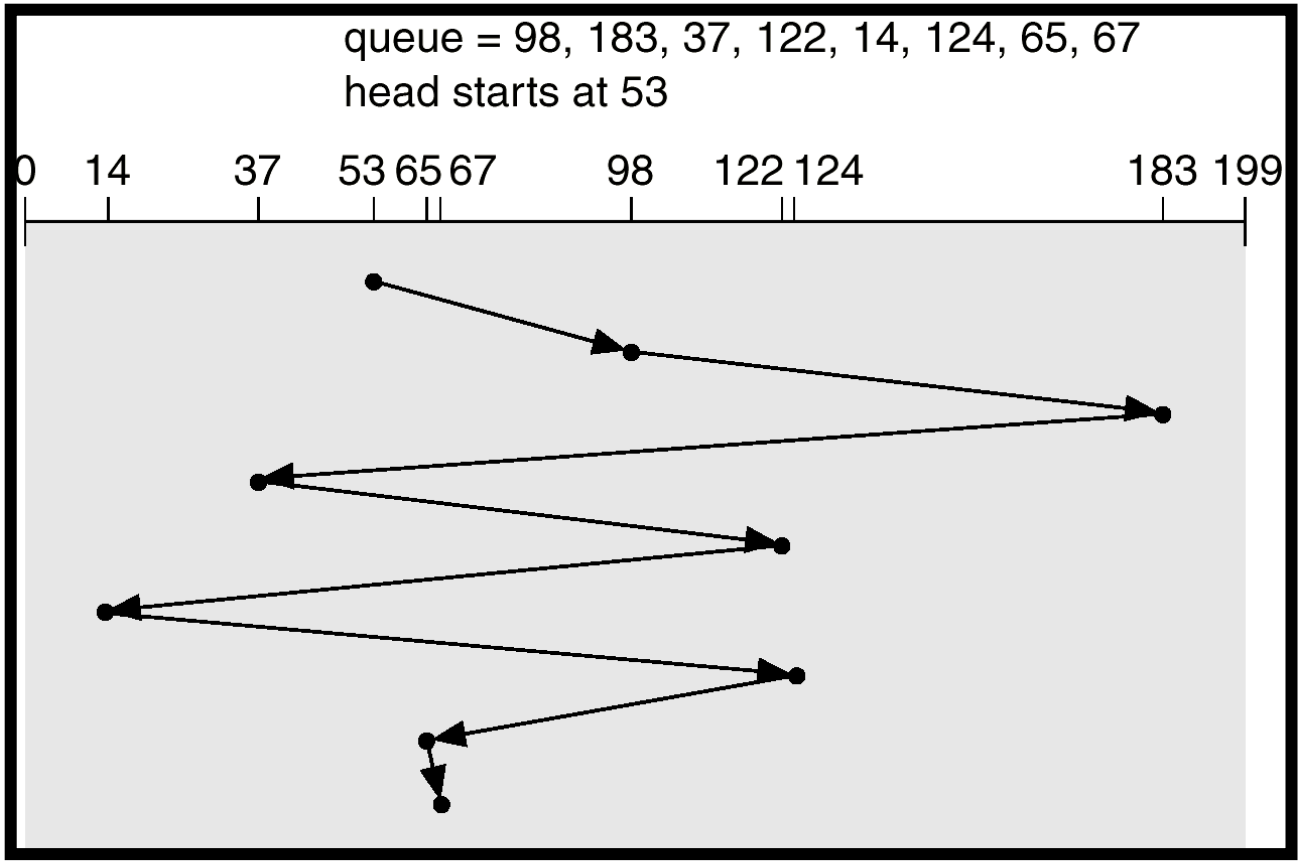
Escalonamento de acesso ao disco

- O “Seek time” é o principal factor no tempo de acesso ao disco
- Para cada disco há um certo número de pedidos de acesso pendentes
- A reordenação da fila pode conduzir a tempos médios de acesso mais pequenos do que os conseguidos com FCFS (first come first served)
- As métricas que definem a qualidade dos algoritmos são:
  - Número de pistas atravessadas pela cabeça
  - Número de inversões de sentido de deslocamento

Escalonamento dos pedidos de acesso ao disco

- A seguir ilustram-se dois algoritmos para escalonar (ordenar) a execução de pedidos de I/O sobre o disco
- Os algoritmos são ilustrados para:
  - um disco com pistas de 0 a 199
  - sequência de pedidos para as pistas 98, 183, 37, 122, 14, 124, 65, 67
  - posição corrente da cabeça: pista 53

Por ordem de chegada

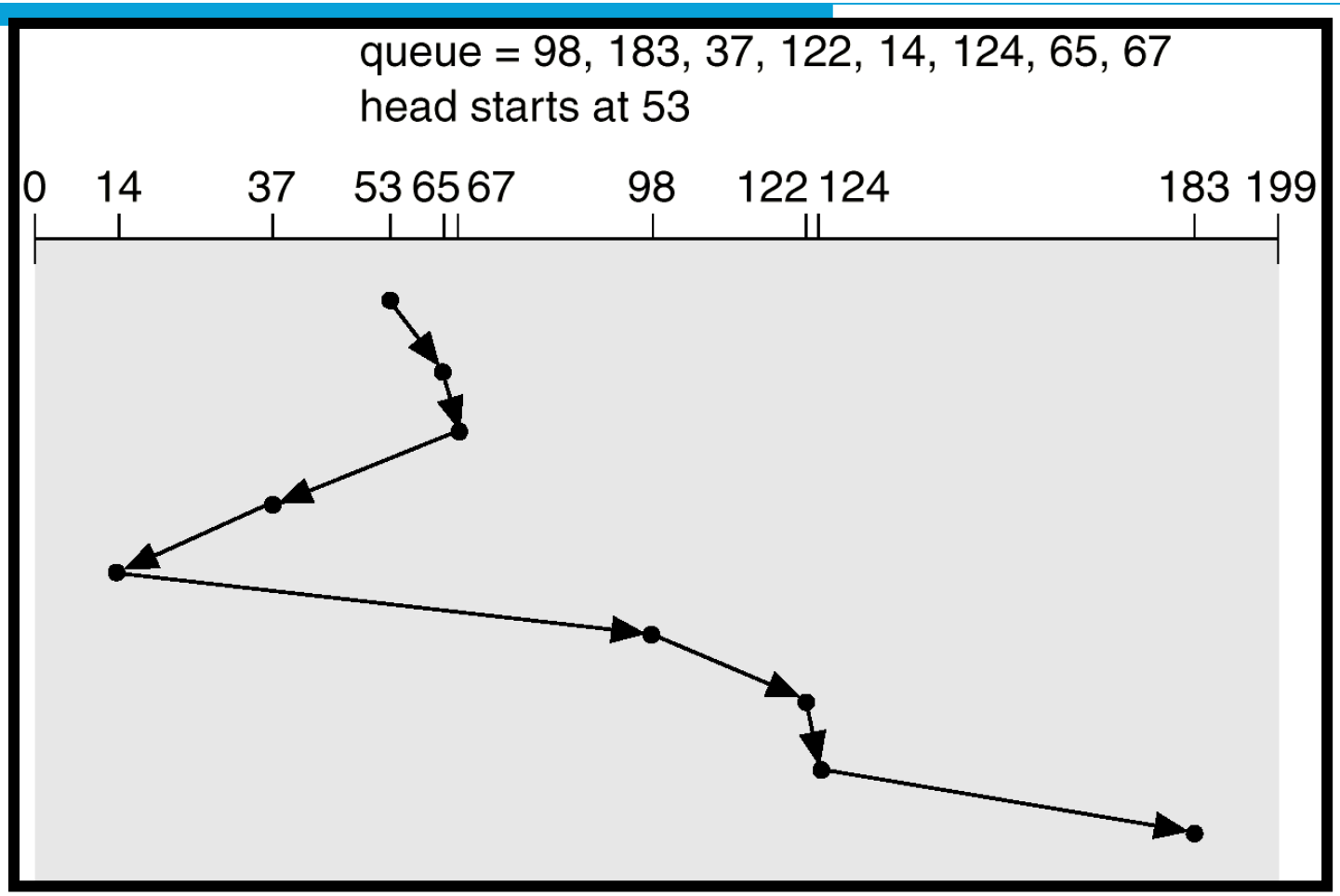


A cabeça movimenta-se sobre um total de 640 cilindros

SSTF (shortest seektime first)

- Selecciona o pedido que corresponde à pista mais próxima da posição corrente da cabeça.
- SSTF pode causar esperas muitas longas a alguns pedidos.

SSTF (Cont.)



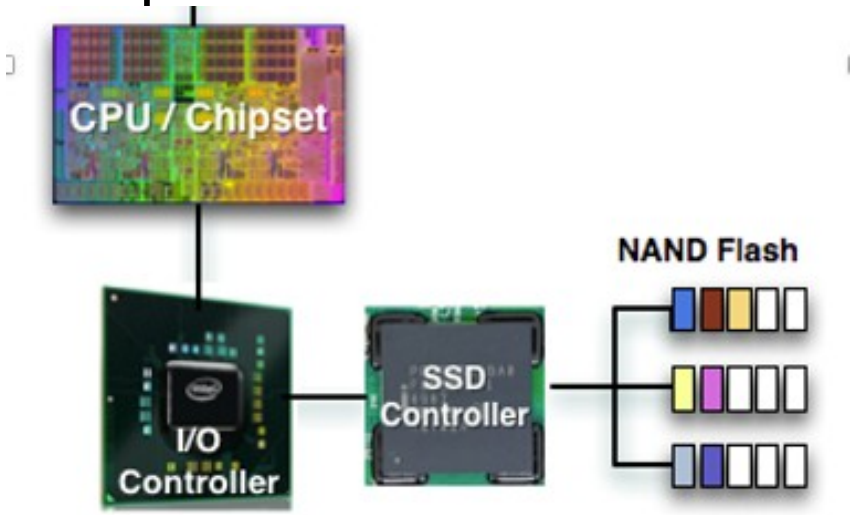
No exemplo o total de movimentos da cabeça é 236 cilindros

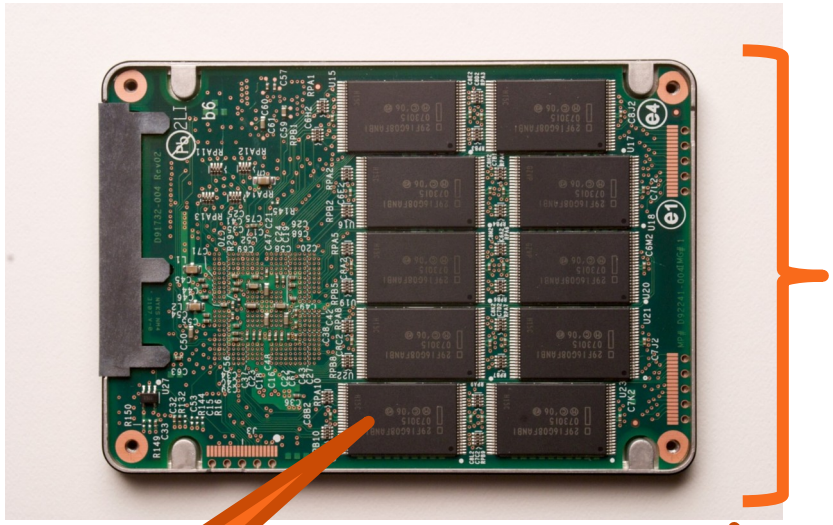
SSD (Solid State Disks)

- Os discos de tecnologia magnética (HD) existem desde 1956
  - Ainda são a forma mais barata de armazenar dados
  - A capacidade continua a aumentar de forma sustentada
- São lentos quando comparados com o CPU e até com alguns periféricos (rede)
- Têm alguns problemas para serem integrados em portáteis, tablets e smartphones
  - tamanho
  - Fragilidade mecânica
  - Consumo

Solid State Drives

- Baseados na tecnologia *NAND flash memory*
  - Armazena-se uma carga elétrica num transistor que representa um ou mais bits





Flash memory chip

Bits espalhados pelos vários *memory chips*

Vantagens dos SSDs

- Mais resistentes a danos físicos
  - Não têm partes móveis (cabeças R/W)
  - São imunes a mudanças de temperatura
- Consomem menos que os HD
- Mais rápidos que os HD
  - >500 MB/s vs ~200 MB/s para HD
  - Não têm problemas com escritas aleatórias
    - Cada “flash cell” pode ser endereçada diretamente
    - Não há “seek time” nem latência rotacional
  - Débito muito elevado
    - Os “flash chips” transferem em simultâneo

Inconvenientes dos SSDs

- Custo por bit
- Controladores complexos por causa da organização da *flash memory*
- Cada flash cell só pode ser alterada um número limitado de vezes (~10000)

Dificuldades com a memória Flash

- Na memória Flash a escrita é ao nível da ***página***, mas para apagar é preciso apagar um **bloco**
  - Páginas: 4 – 16 KB, Blocos: 128 – 256 KB
  - Problema amplificação das escritas (write amplification)

    Para que se escreva uma página pode haver necessidade de ler apagar e reescrever um ou mais blocos

- A memória Flash tem um nº limite de vezes que pode ser alterada
  - Típico 3000 – 5000 vezes
  - Isto obriga a mudar os dados de posição para distribuir as escritas por todas as células de forma equilibrada (wear leveling) por todas as células
  - Com isto, pretende-se garantir uma vida útil de 5 anos

Controladores dos SSD

- SSD são bastante complicados internamente
- O controlador do SSD
  - Mapeia o Nº do bloco para uma página física
  - Gere um mapa de páginas livres e sua relação com os blocos
  - Pode, em background, realizar compactação de espaço (*garbage collection*) procurando maximizar o nº de blocos com todas as páginas livres
  - Faz *wear leveling* rodando os blocos lógicos dos ficheiros por diferentes blocos do SSD

Dois tipos de Flash Memory

Multi-Level Cell (MLC)

- Mais de um bit por flash cell
  - 2 níveis: 00, 01, 10, 11
  - Há MLC de 2, 3 e 4 bits
- Mais capacidade e menor preço por bit que as SLC flash
- Menor velocidade de leitura e escrita
- 3000 – 5000 ciclos de escrita
- Maior consumo

Usadas em portáteis, …

Single-Level Cell (SLC)

- Um bit por flash cell
  - 0 or 1
- Menor capacidade e maior custo / bit que a MLC flash
- Maior velocidade de leitura e escrita do que as MLC
- 10000 – 100000 ciclos de escrita

Usados em servidores de desempenho elevado

Estruturas RAID

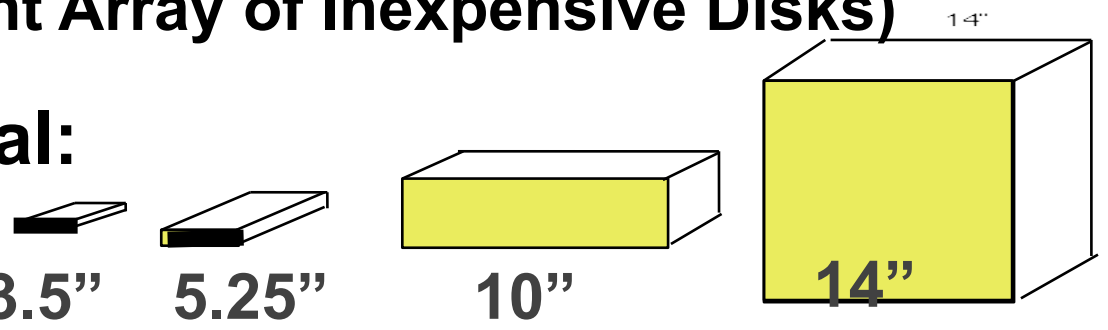
- **RAID** – múltiplas unidades de disco suportam elevada **fiabilidade** através de **redundância**.
- Inicialmente foram definidos 6 níveis RAID
  - RAID 0 : só assegura aumento da velocidade de acesso, porque permite acessos em paralelo
  - RAID 1, 2, 3, 4 e 5: permite tolerância a falhas, porque há discos extra que asseguram redundância
- Outros níveis definidos mais recentemente
  - 6, 10 (ou 1+0) …
- A maior parte das vezes é um sistema complexo (exterior à caixa do computador) conhecido por **disk array**
  - Com um Sistema operativo dedicado; memória não-volátil,…

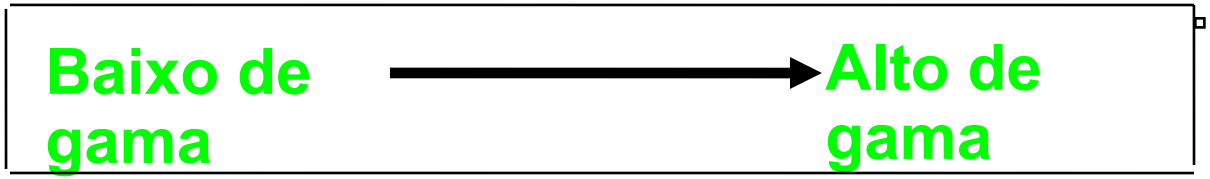
Proposta inicial (Patterson 1988)

SLED (Single Large Expensive Disk)
VS
RAID (Redundant Array of Inexpensive Disks)

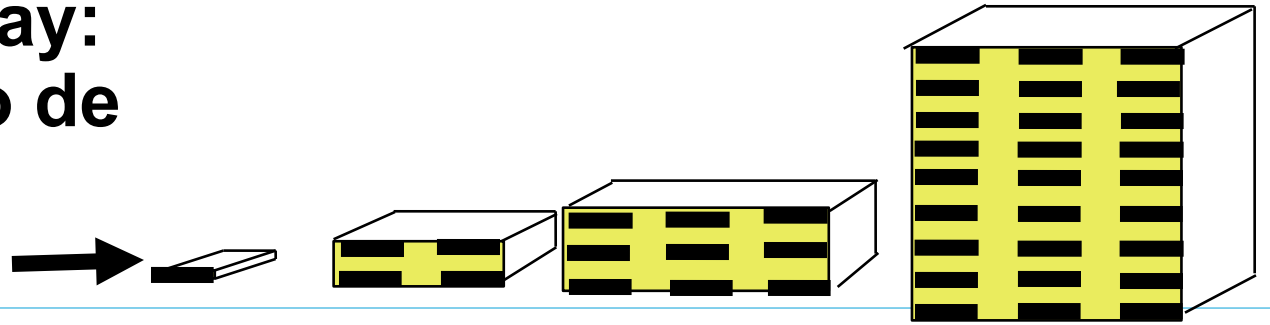
Convencional:

4 tipos de discos





Disk Array:
1 só tipo de disco



Substituir um grande disco por muitos pequenos discos! (Patterson 1988)

	IBM 3390 (K)	IBM 3.5" 0061	*x70*
Capacidade	20 GBytes	320 MBytes	*23 GBytes*
Volume	97 cu. ft.	0.1 cu. ft.	*11 cu. ft.*
Potência	3 KW	11 W	*1 KW*
Ritmo de transf	15 MB/s	1.5 MB/s	*120 MB/s*
Ritmo de ops I/O	600 I/Os/s	55 I/Os/s	*3900 IOs/s*
MTTF	250 KHrs	50 KHrs	*??? Hrs*
Custo	$250K	$2K	*$150K*

Disk Arrays têm potencial para

- Grandes taxas de transferência
- high MB por cu. ft., high MB por KW
- fiabilidade?

Fiabilidade do Array

- **Fiabilidade de N discos = Fiabilidade de 1 Disco ÷ N**

  50,000 Horas ÷ 70 discos = 700 horas

  MTTF do sistema de discos : Desce de 6 anos para 1 mês!

  • Arrays (sem redundância) demasiado pouco fiáveis para terem utilidade!

Suporte de "hot swap" com reconstrução em paralelo com a operação normal permite tem uma disponibilidade extremamente elevada

RAID=Redundant Arrays of Independent Disks

- **Os blocos são distribuídos ("striped“) por vários discos**
- **Redundância garante alta disponibilidade dos dados**

Em caso de falha de um disco, o conteúdo é reconstruído a partir de dados redundantes armazenados no array

→ **Perde-se capacidade para os armazenar**

→ **Existe uma penalização em bandwidth para actualização**

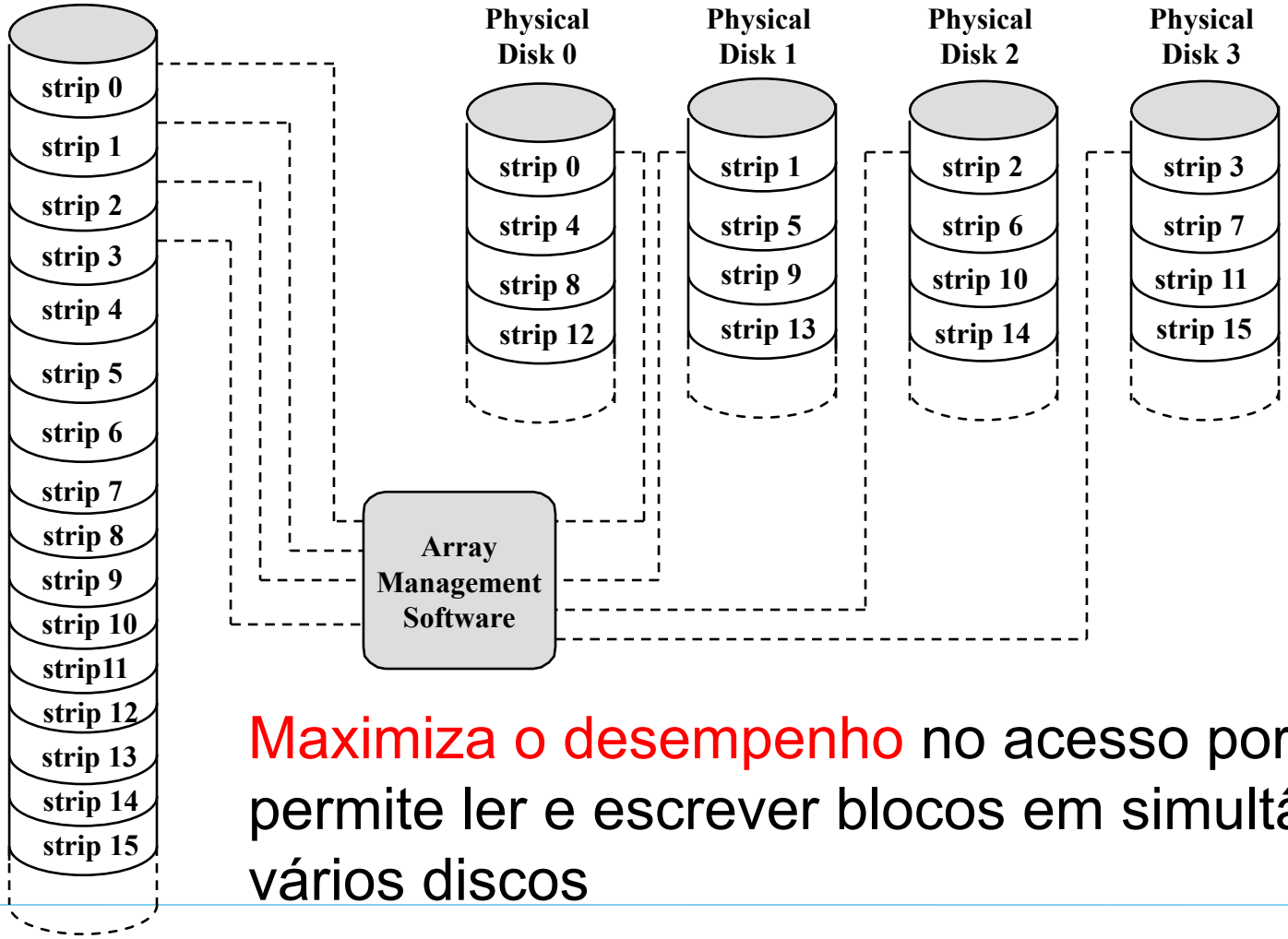
Técnicas:

- **Mirroring/Shadowing (elevado custo em capacidade)**
- **Códigos de Correção de Erros (paridade, outros)**

RAID (cont)

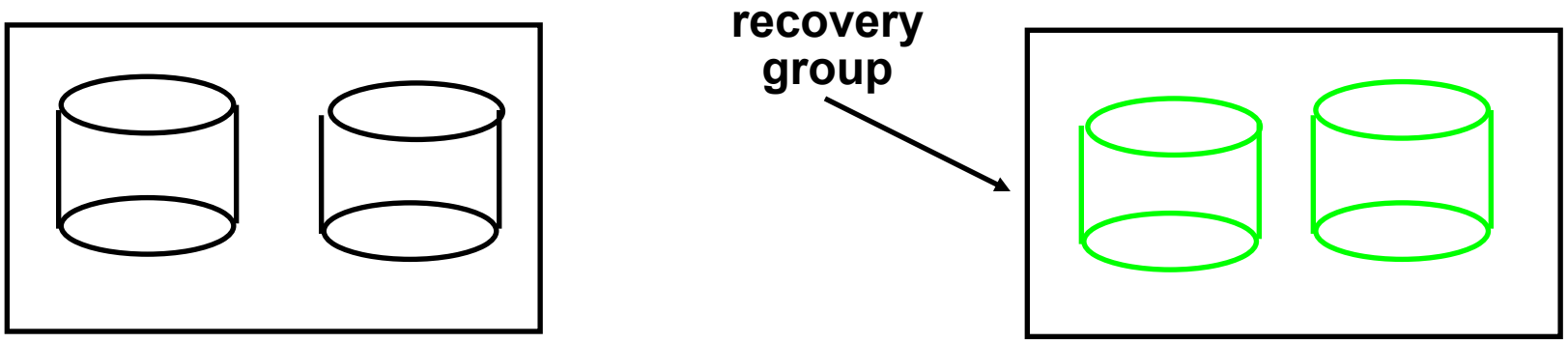
- Duas técnicas usadas:
  - “Disk striping” usa um grupo de discos como uma unidade lógica: diferentes partes dos dados são armazenados em discos diferentes
  - Aumento da velocidade de acesso e da fiabilidade através do armazenamento de dados redundantes.
    - *Mirroring* ou *shadowing* duplica discos inteiros.
    - *Block interleaved parity* usa muito menos redundância.

RAID Nível 0 (não redundante)



Maximiza o desempenho no acesso porque permite ler e escrever blocos em simultâneo vários discos

RAID 1
Disk Mirroring/Shadowing

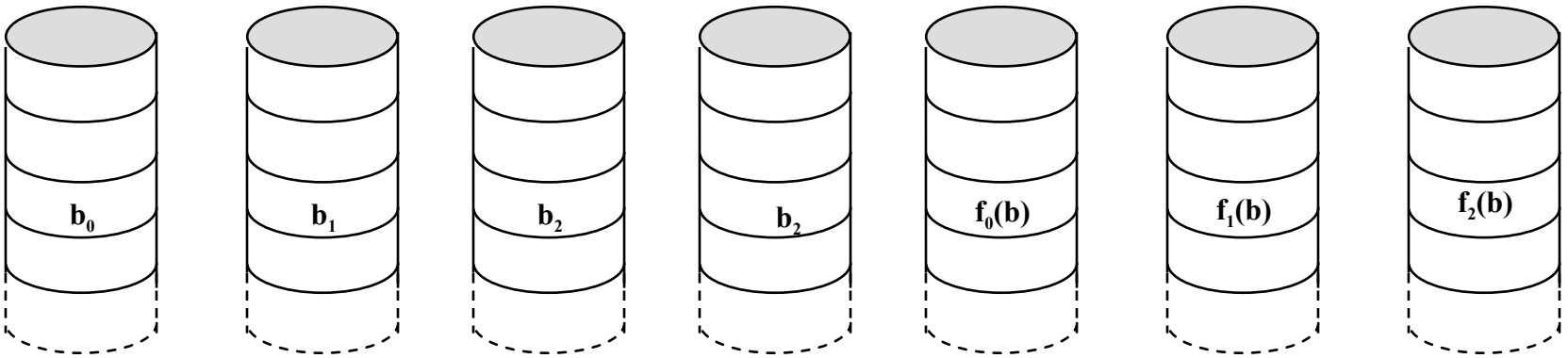


- **Cada disco é completamente duplicado no seu "shadow"**
  Pode ser atingida uma disponilidade muito elevada
- **Sacrifício da largura de banda em escrita:**
  uma escrita lógica = duas escritas físicas
- **As leituras podem ser optimizadas**
- **A solução mais cara: 100% de custos extra em capacidade**

Usado em ambientes em que interessa alta disponibilidade

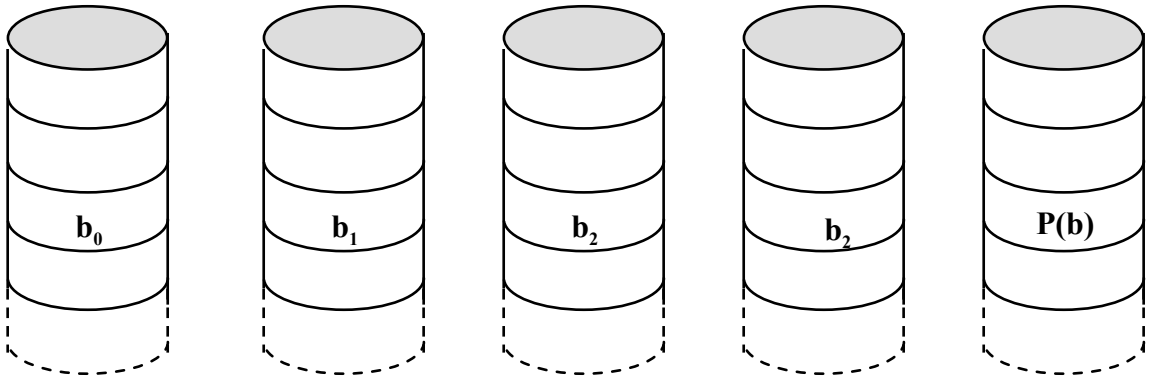
RAID 2 e 3

Códigos de correção de erros ao nível do bit



Praticamente, não são usados

Exemplo de correção de erros



O conteúdo do disco i pode ser obtido fazendo

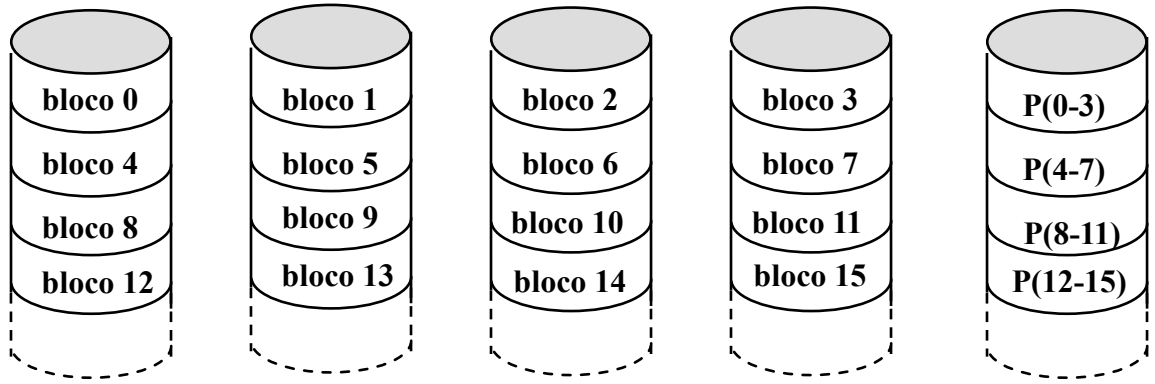
$b_i = b_0$ XOR b_1 XOR ... b_{j-1} XOR B_j (j <> i)

(paridade par)

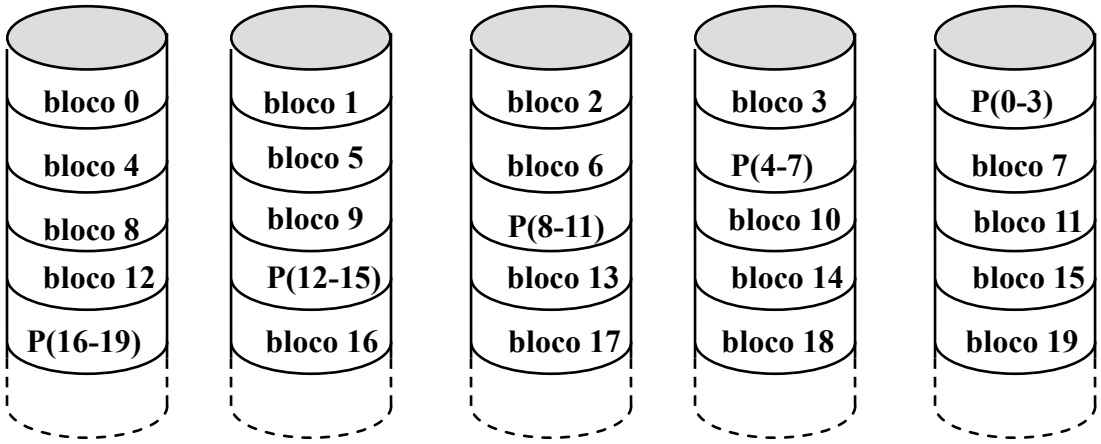
Isto permite:

- Operar com o disco i estragado
- Introduzir o disco i e refazer o seu conteúdo

RAID 4 (paridade a nível do bloco)



RAID 5 (paridade a nível do bloco distribuída)



Escritas em RAID 5

- RAID 5 é um bom compromisso velocidade / espaço desperdiçado para redundância
- Em RAID 5 uma operação de escrita inclui:
  - Escrever um novo conteúdo Y um bloco de dados D que continha X:
    - Ler o bloco D
    - Ler o bloco de paridade correspondente com o conteúdo Z
    - Escrever Y em D
    - Se fizermos (X xor Y) temos a 1 os bits diferentes
    - o novo conteúdo do bloco é Z xor (X xor Y); troca os bits de paridade nas posições em que há diferenças
  - 2 leituras e duas escritas, mas pode haver paralelismo entre as duas leituras e as duas escritas