Arquitetura de Computadores III

Hierarquia de memória
Memória virtual

Introdução

- memória principal semicondutora
  - capacidade limitada
  - tempo de acesso entre 10 e 20 ns
- memória secundária em disco
  - Capacidade limitada (obviamente) mas muito maior
  - tempo de latência entre 10 e 30 ms

O problema

- Um computador tem 32 kbytes de memória principal

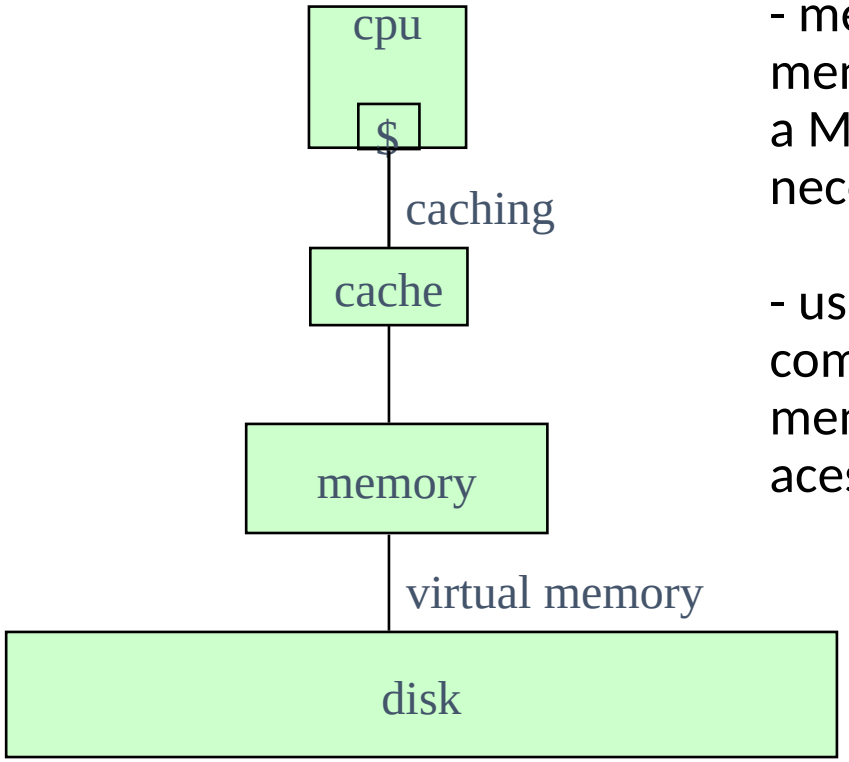
- Como podemos:
  - rodar programas que usam mais do que 32 kbytes?
  - permitir que vários usuários usem o computador?
  - executar vários programas ao mesmo tempo?

- O problema é independente do tamanho da memória. Pode trocar para 32 MB ou 32 GB, que o problema permanece. A memória é dimensionada para atender a carga de trabalho típica do sistema.

Memória virtual: a solução

- *Memória Virtual* : técnica que nos permite ver a memória principal como uma cache de grande capacidade de armazenamento

- É apenas mais um nível na hierarquia de memórias



- mecanismo automático de gerência de memória, que traz automaticamente para a MP os blocos de informação (do disco) necessários

- usuário tem a impressão de trabalhar com uma memória única, do tamanho da memória secundária, mas com tempo de acesso próximo do tempo da MP

Tempo de acesso

Tempo médio de acesso Tma é dado por

$$Tma = Tm + (1 – h) Ts$$

onde Tm = tempo de acesso à MP
Ts = tempo de acesso ao disco
h = hit ratio

p.ex. se Tm = 20 ns, Ts = 20 ms, h = 0.9999
então Tma = 2,02 µs (100 x maior do que Tm)

Por que MV é diferente das caches?

- Miss penalty é MUITO maior (milhões de ciclos)! Se informação não está na memória, está no disco!

- Logo:
  - miss ratio precisa ser bem menor do que em cache
  - alta penalidade do miss => necessário buscar blocos maiores em disco
  - princípio de localidade opera sobre blocos maiores de dados ou instruções e leva a hit ratios bem mais elevados
  - Mapeamento totalmente associativo das páginas
  - misses são tratados por software (há tempo disponivel)
  - técnica de escrita write-through não é uma opção. Usa-se write-back.

Terminologia

- mesma ideia da cache, mas com terminologia diferente

cache	MV
bloco	página
cache miss	*page fault*
endereço	endereço virtual (ou lógico)
índice	endereço real (ou físico)

- endereço *virtual (lógico)*: gerado pelo programa
  - deve endereçar todo espaço em disco
  - maior número de bits

- endereço *real (físico)*: endereço na memória principal
  - menor número de bits

Unidade de Gerenciamento de Memória

- MMU (Memory Management Unit)
  - gerência da hierarquia de memória
  - proteção de memória
  - usualmente integrada dentro do microprocessador
- MMU deve fazer mapeamento do endereço virtual para endereço real
- SO usa a MMU

Paginação (gerenciamento de memória)

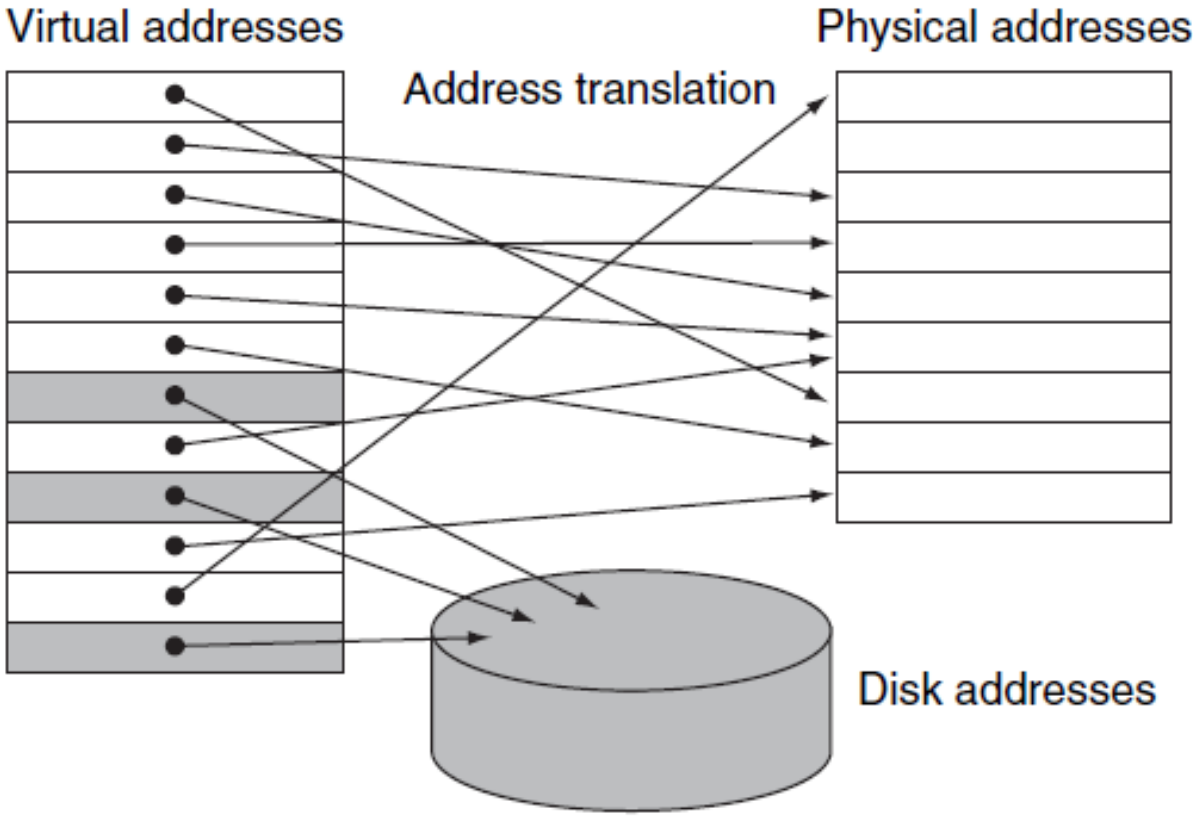
- Por que paginação? Resposta: mecanismo simples para tradução de endereços virtuais em reais e para gerenciamento do espaço de memória

- espaços de memória real e virtual divididos em blocos chamados de páginas
  - páginas tem tipicamente de 4 kbytes a 16 kbytes
  - Páginas para sistemas embarcados são de 1 kbytes

- endereços virtuais e reais divididos em 2 campos
  - endereço da página
  - endereço da linha (ou palavra), dentro da página

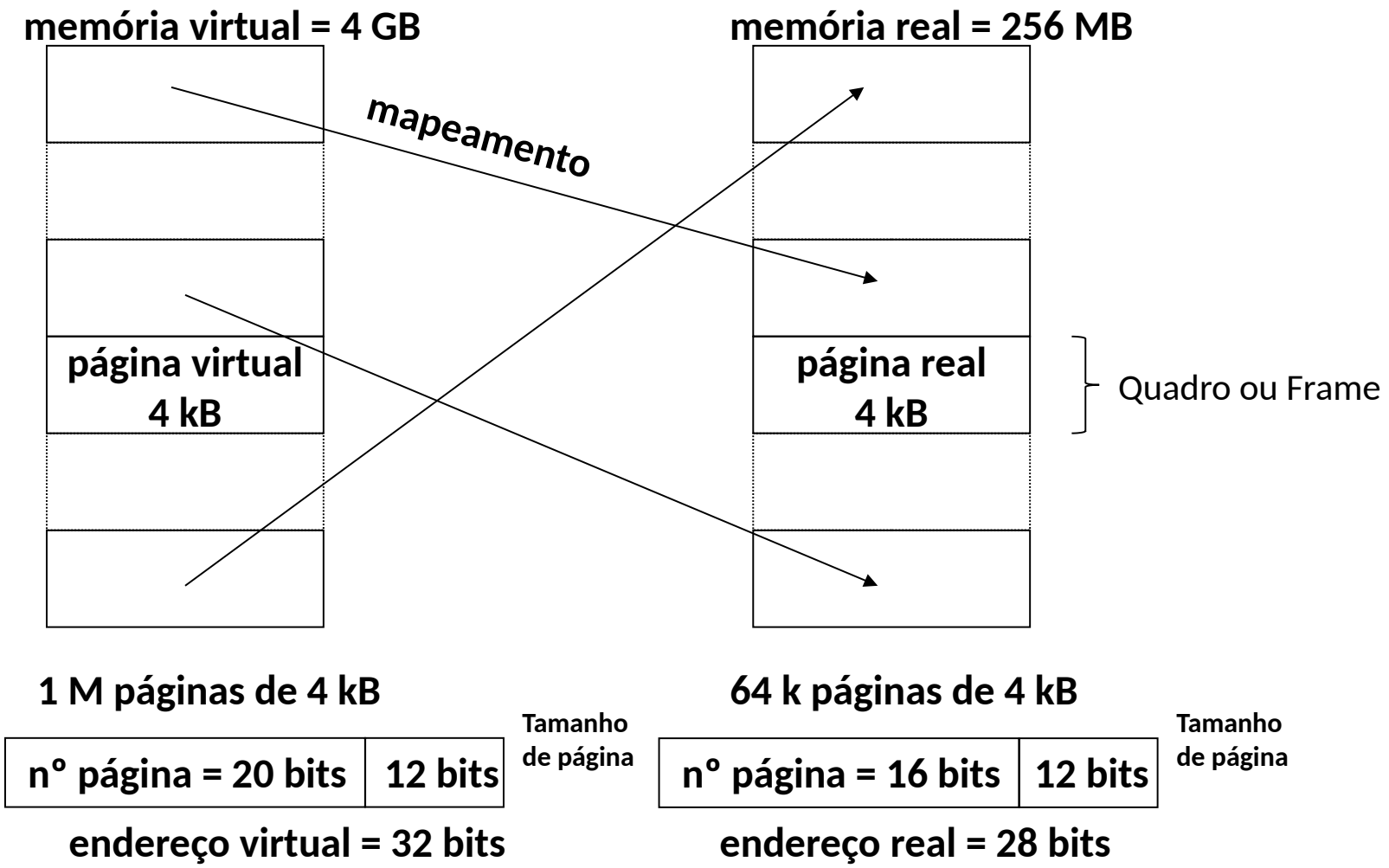
Paginação

- É apenas uma função de mapeamento dos endereços virtuais (do disco) para endereços reais (físicos) na memória principal



PATTERSON, David A. Organização e projeto de computadores a interface hardware/software

Paginação


memória virtual = 4 GB
memória real = 256 MB
mapeamento
página virtual
4 kB
página real
4 kB
Quadro ou Frame
1 M páginas de 4 kB
64 k páginas de 4 kB
nº página = 20 bits
12 bits
Tamanho de página
nº página = 16 bits
12 bits
Tamanho de página
endereço virtual = 32 bits
endereço real = 28 bits

Paginação

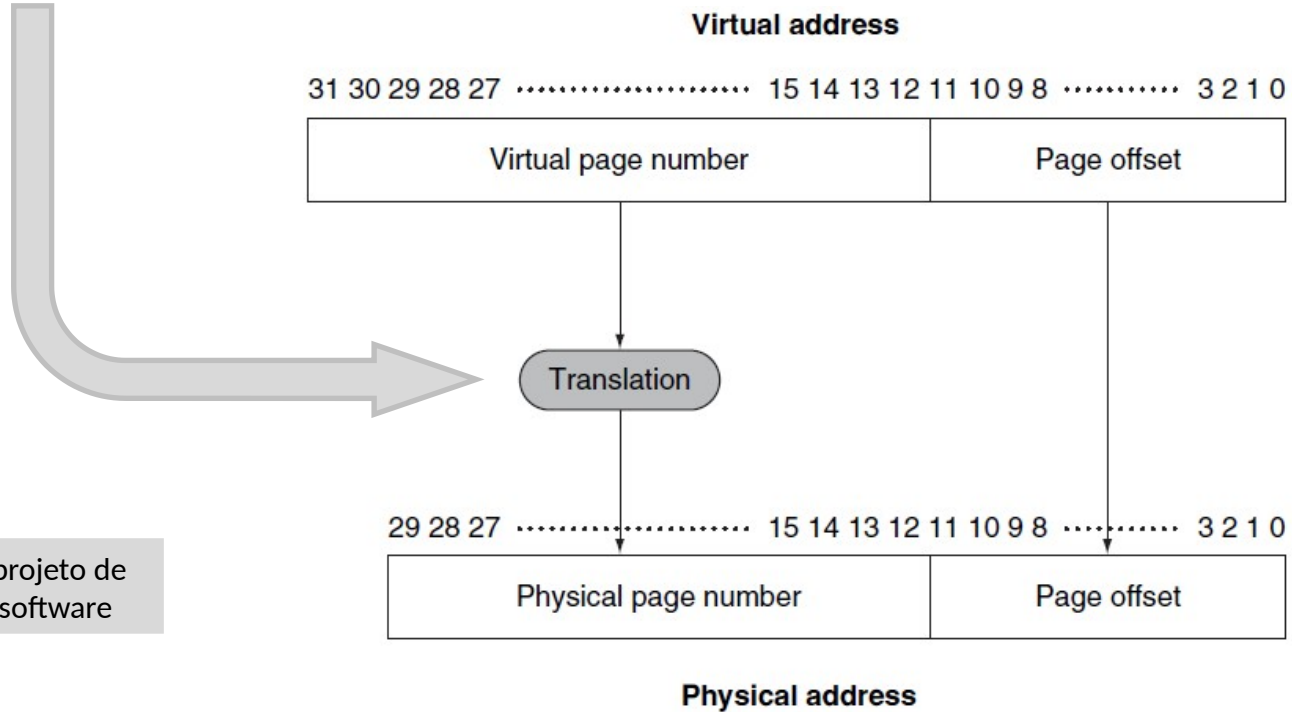
- Page fault ocorre quando a página virtual não está na memória principal
- Mapeamento completamente associativo, mais eficiente, ajuda a diminuir alta penalidade dos page faults

- Como transformar endereçamento original do programa no endereçamento real?

- page tables
  - guardam a correspondência entre páginas virtuais e páginas reais
  - permitem a translação de endereços

Paginação

- Como transformar endereçamento original do programa no endereçamento real?
- Page tables
  - guardam a correspondência entre páginas virtuais e páginas reais
  - permitem a translação de endereços



PATTERSON, David A. Organização e projeto de computadores a interface hardware/software

Gerência de processos

- cada processo tem sua própria tabela de páginas
  - processos são compilados para espaços de endereçamento virtuais
  - tabela de páginas define toda a utilização do espaço de endereçamento pelo processo

- sistema operacional é responsável pela alocação de espaço físico para o espaço virtual de cada processo
  - SO carrega tabela de páginas de cada processo

- hardware possui registrador que aponta para início da tabela de páginas do processo atual

- quando novo processo passa a ser ativo, sistema operacional só precisa atualizar valor deste registrador

Paginação

- main memory translation table (MMTT)
  - implementada em hardware (*em tese*)
  - tamanho = nº de páginas na memória principal

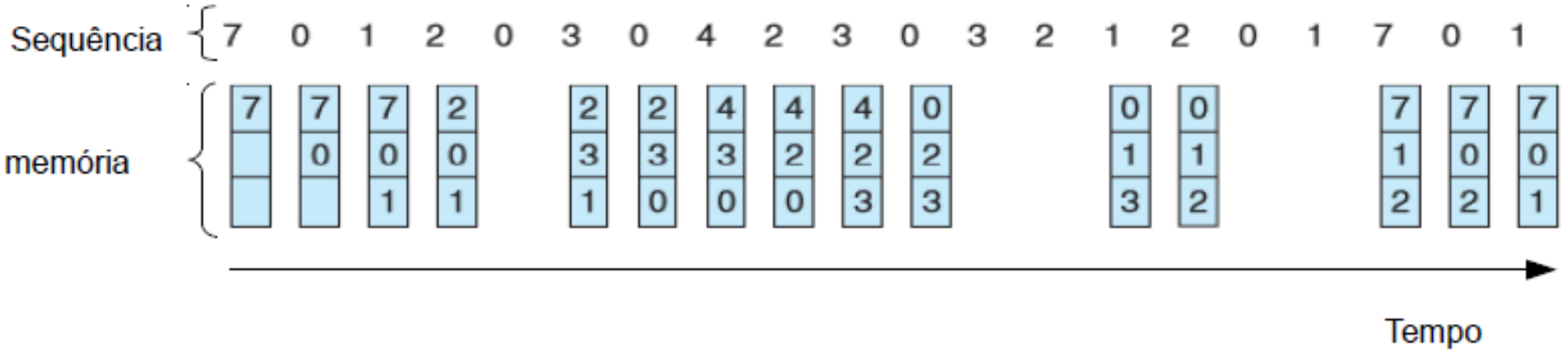
- disk memory translation table (DMTT)
  - implementada em software, armazenada na memória principal
  - tamanho = nº de páginas em disco

- **algoritmo de substituição**, em software, para selecionar página da memória principal a ser substituída em caso de page fault

- bom desempenho é garantido pelo princípio de localidade

Substituição de páginas

- FIFO
- Memória com 3 frames (quadros)
  - Frame tem tamanho igual a uma página virtual

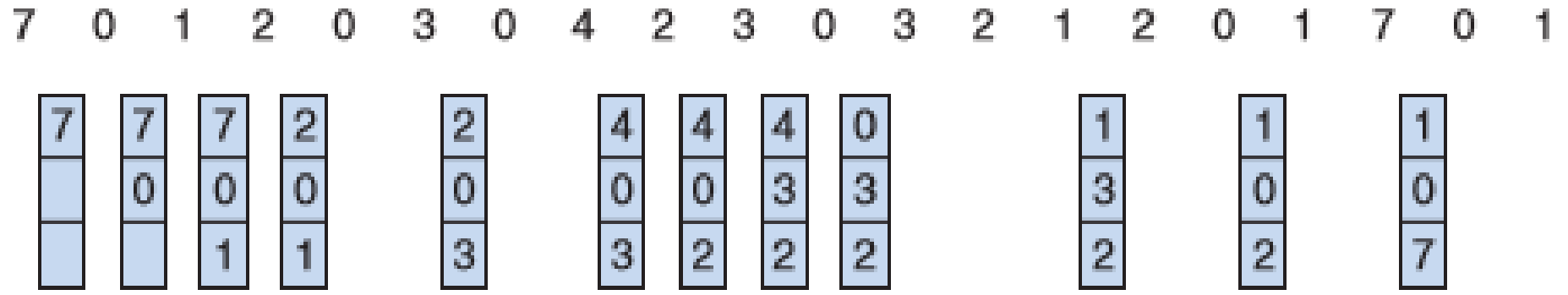


- Total de falhas = 15
- Taxa de falha = 15/20 = 0,75

https://www.cs.uic.edu/~jbell/CourseNotes/OperatingSystems/9_VirtualMemory.html

Substituição de páginas

- LRU
- Memória com 3 frames (quadros)
  - Frame tem tamanho igual a uma página virtual

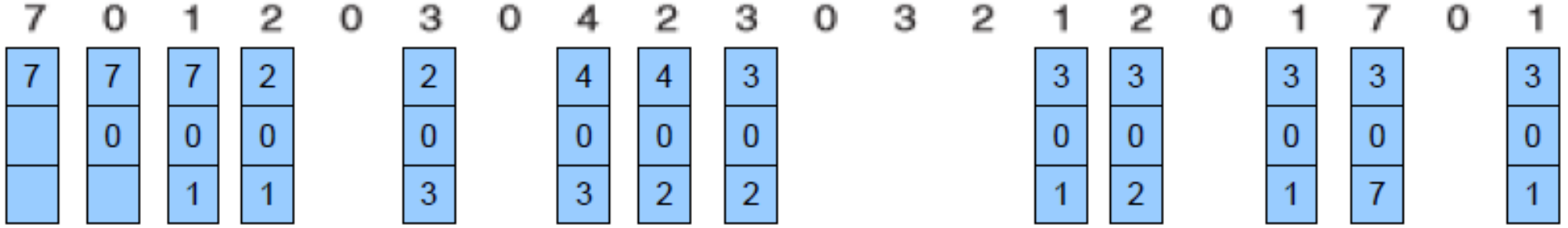


  - Total de requisições = 20
  - Total de falhas = 12
  - Taxa de falha = 12/20 = 0,6

https://www.cs.uic.edu/~jbell/CourseNotes/OperatingSystems/9_VirtualMemory.html

Substituição de páginas

- LFU
  - FIFO como critério de desempate
- Memória com 3 frames (quadros)
  - Frame tem tamanho igual a uma página virtual

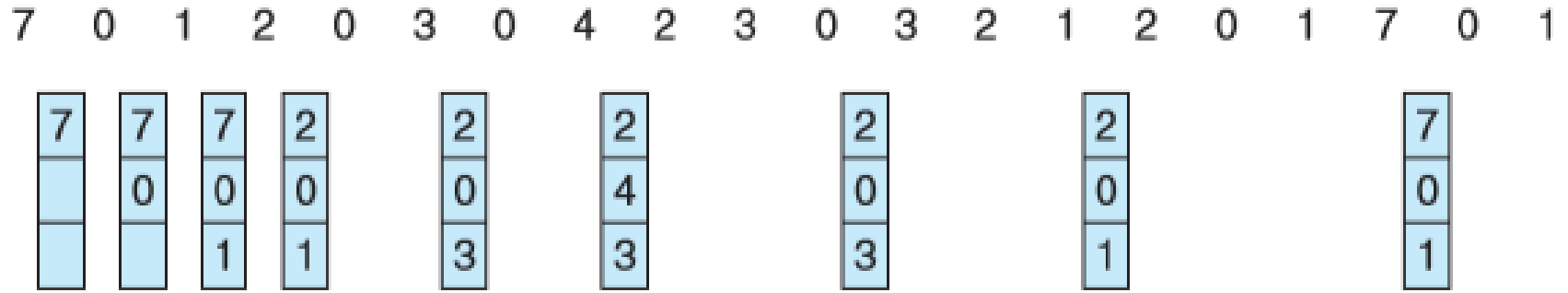


  - Total de requisições = 20
  - Total de falhas = 13
  - Taxa de falha = 13/20 = 0,65

https://www.cs.uic.edu/~jbell/CourseNotes/OperatingSystems/9_VirtualMemory.html

Substituição de páginas

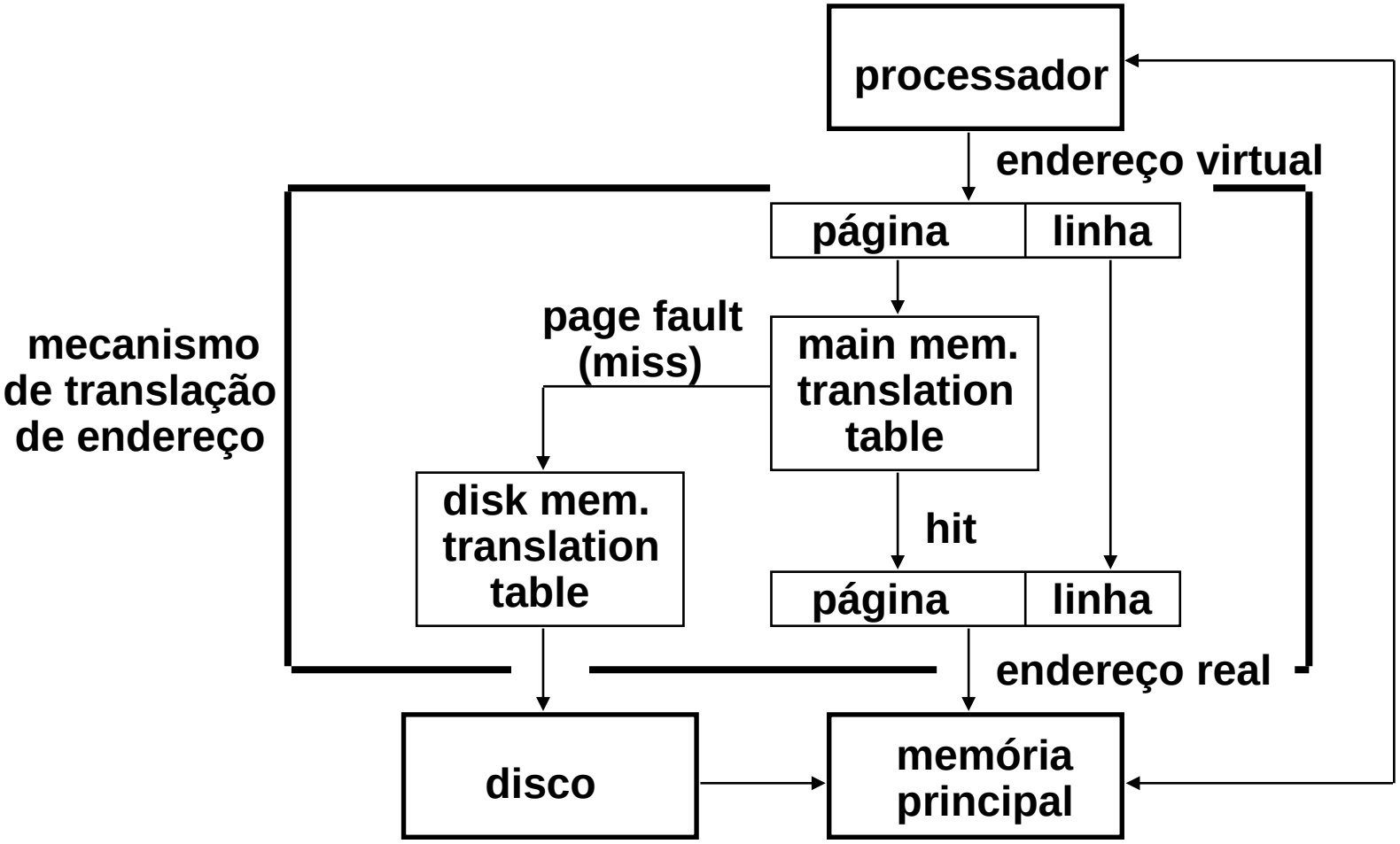
- OPT (solução ótima)
  - Substitui a página que não será usada por um tempo mais longo (prever futuro)
- Memória com 3 frames (quadros)
  - Frame tem tamanho igual a uma página virtual



  - Total de requisições = 20
  - Total de falhas = 9
  - Taxa de falha = 9/20 = 0,45

https://www.cs.uic.edu/~jbell/CourseNotes/OperatingSystems/9_VirtualMemory.html

Paginação (translação do endereço)



Translation Look-Aside Buffer

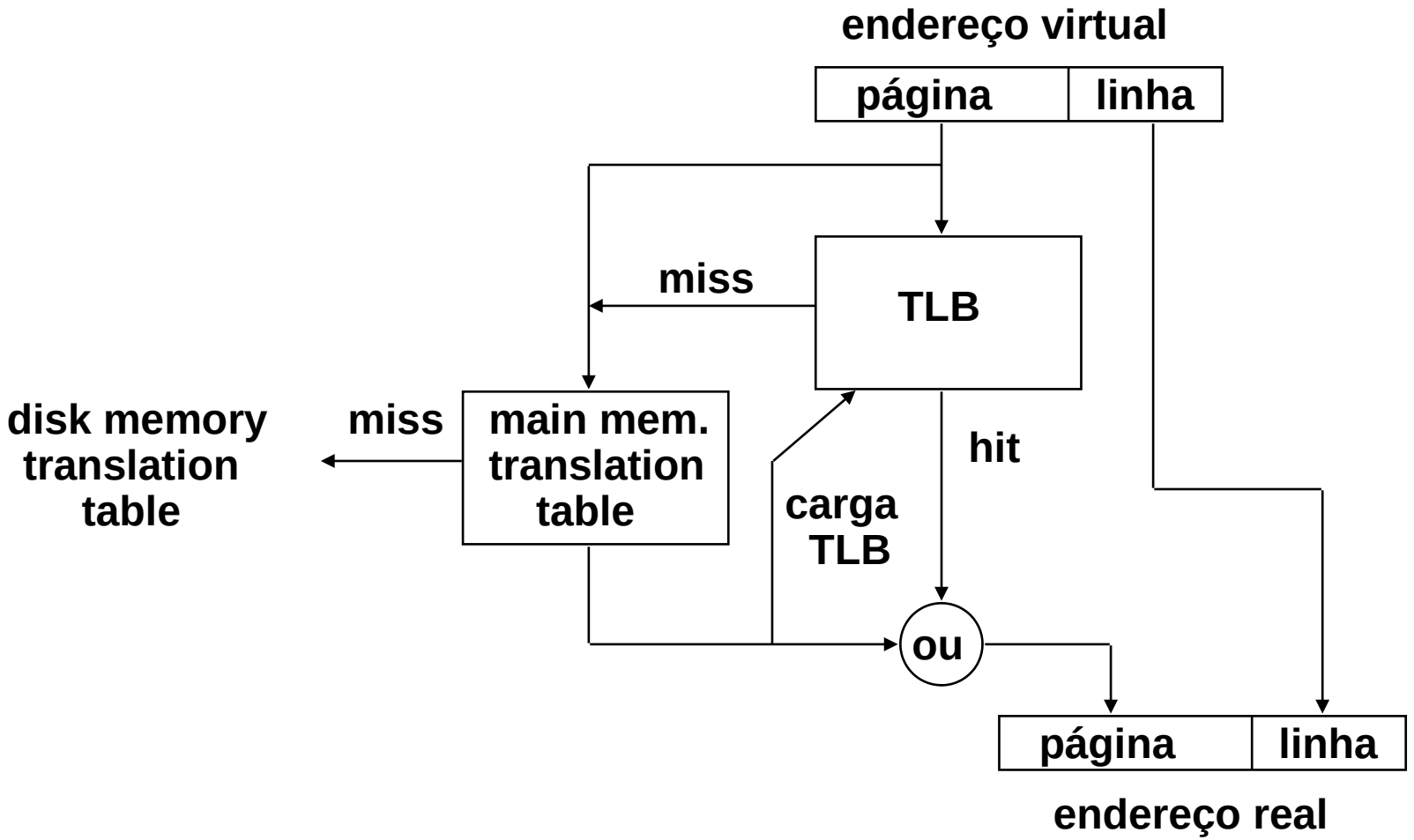
- nº de páginas na memória secundária é muito grande
  - espaço virtual de 2^{32}bytes, páginas de 4k bytes, 4 bytes por entrada na tabela
  - 4 MBytes apenas para a tabela de páginas!!!
  - tamanho excessivo da main memory translation table
- se tabela ficar na memória principal => dois acessos à memória a cada cache miss

- working set = conjunto de páginas mais prováveis de serem acessadas num dado momento, devido ao princípio de localidade

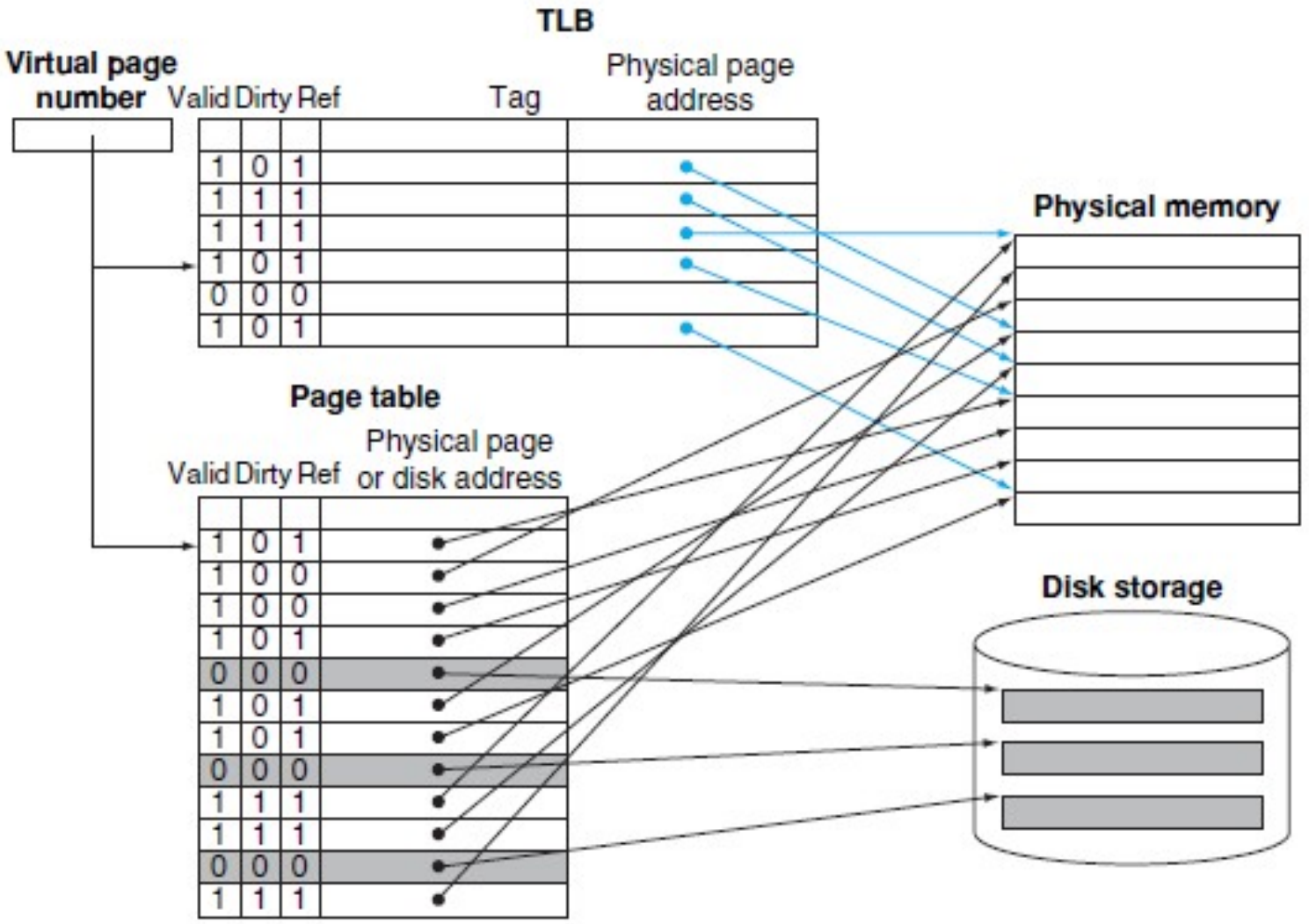
- Translation Look-Aside Buffer (TLB)
  - implementado em hardware
  - traduz endereços virtuais para endereços reais
  - só inclui páginas do working set
  - pode ser considerado como uma “cache” da MMTT

- Main Memory Translation Table (MMTT)
  - implementada em **software**

Translation Look-Aside Buffer

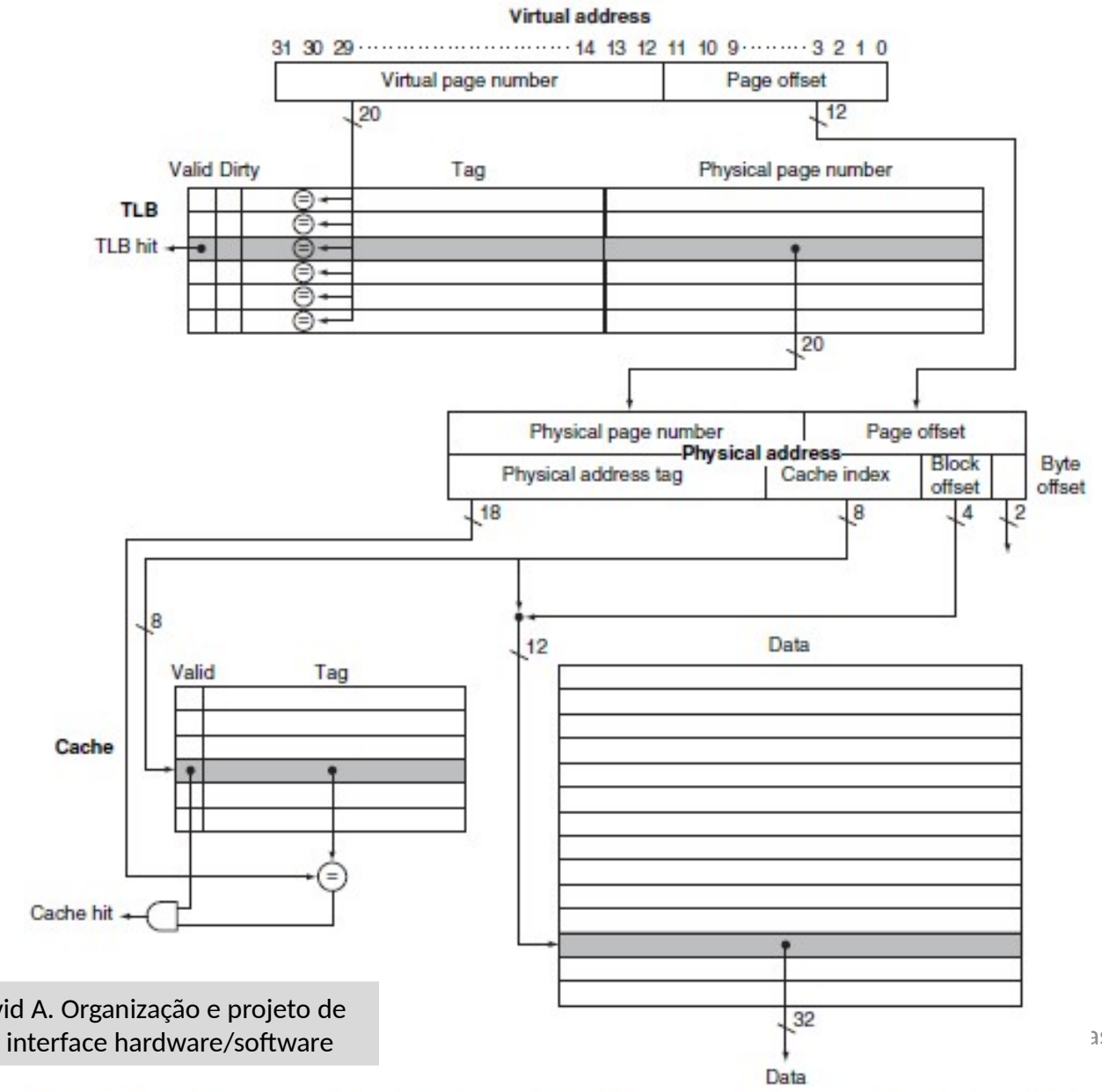


Translation Look-Aside Buffer



PATTERSON, David A. Organização e projeto de computadores a interface hardware/software

Translation Look-Aside Buffer



Cache física ou Cache virtual?

Qual a diferença?

PATTERSON, David A. Organização e projeto de computadores a interface hardware/software

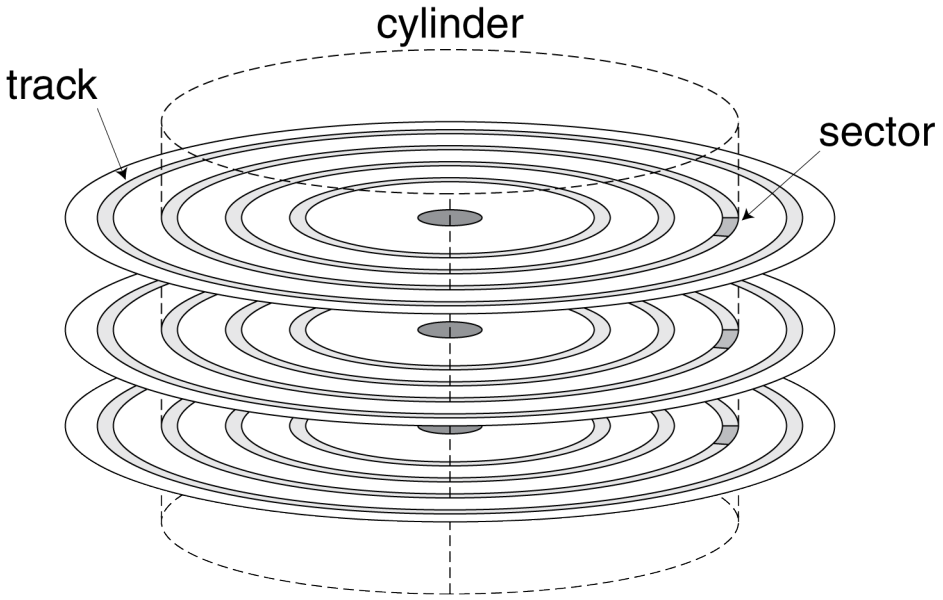
Translation Look-Aside Buffer

TLB	Page table	Cache	Possible? If so, under what circumstance?
Hit	Hit	Miss	Possible, although the page table is never really checked if TLB hits.
Miss	Hit	Hit	TLB misses, but entry found in page table; after retry, data is found in cache.
Miss	Hit	Miss	TLB misses, but entry found in page table; after retry, data misses in cache.
Miss	Miss	Miss	TLB misses and is followed by a page fault; after retry, data must miss in cache.
Hit	Miss	Miss	Impossible: cannot have a translation in TLB if page is not present in memory.
Hit	Miss	Hit	Impossible: cannot have a translation in TLB if page is not present in memory.
Miss	Miss	Hit	Impossible: data cannot be allowed in cache if the page is not in memory.

PATTERSON, David A. Organização e projeto de computadores a interface hardware/software

Disco

- Para acessar os dados:
  - Tempo de busca: posiciona a cabeça sobre a trilha correta (média de 3 a 14 ms)
  - Latência rotacional: espera pelo setor desejado (0,5 rpm)
  - Tempo de transferência: recupera os dados (um ou ou mais setores; ex: 30 a 80 MB/seg)



PATTERSON, David A. Organização e projeto de computadores a interface hardware/software

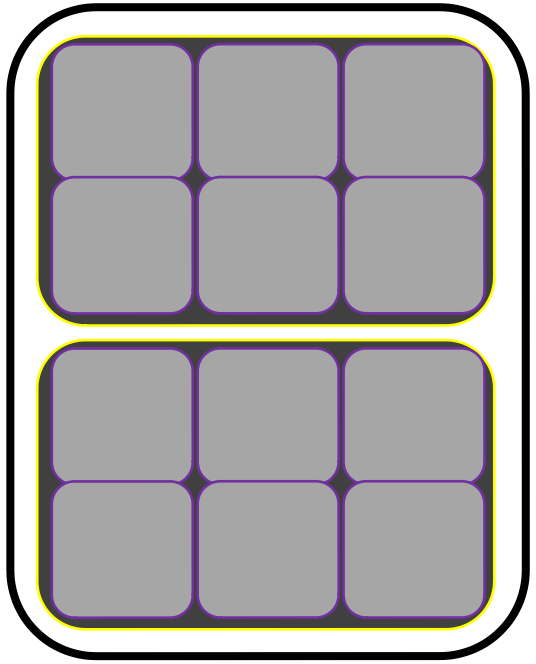
Desempenho do Disco

- Tempo de Disco = Tempo de busca + Latência rotacional + Tempo de transferência
- A latência rotacional média para a informação desejada está a meio caminho ao redor do disco.
- TB = 1ms
- $LRM = \frac{0{,}5 \text{ rotação}}{5400 \text{ RPM}} = \frac{0{,}5 \text{ rotação}}{5400\text{RPM}/60(\text{seg})} = 0{,}0056\text{s} = 5{,}6\text{ms}$
- $TT = 1\text{kB} / 50\text{MB/s} = 1 * 2^{10} / 50 * 10^{6} = 20{,}48\mu\text{s}$
- TD = 1ms + 5,6ms + 0,02048ms = 6,62048ms

Disco de estado sólido (SSD)

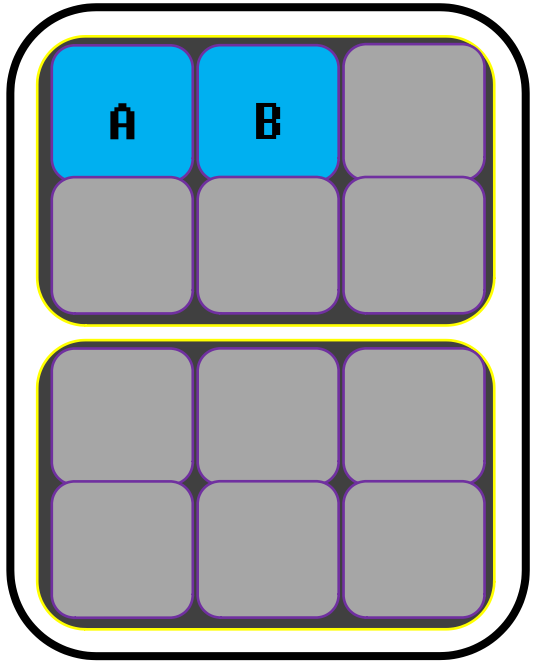
- Utiliza circuitos eletrônicos para armazenamento de longo prazo
  - NAND Flash memory
- Operação eletrônica: mais rápida e sem deslocamentos
- Acesso uniforme a dados
- Potencial perda de confiabilidade após certa quantidade de utilizações

Disco de estado sólido (SSD)



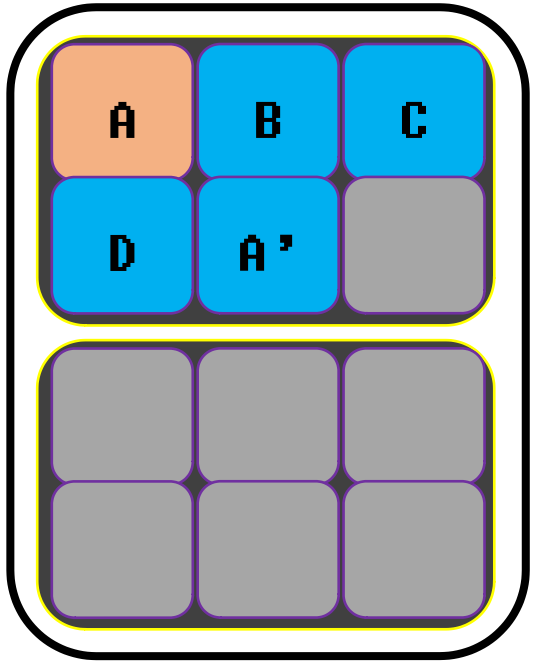
- Operações de:
  - Reescrita
  - Apagamento
  - Coleta de lixo

Disco de estado sólido (SSD)



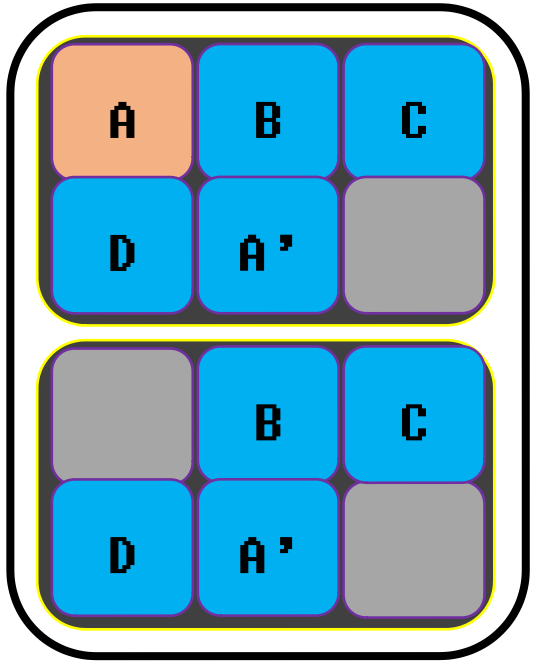
- Operações de:
  - Reescrita
  - Apagamento
  - Coleta de lixo

Disco de estado sólido (SSD)



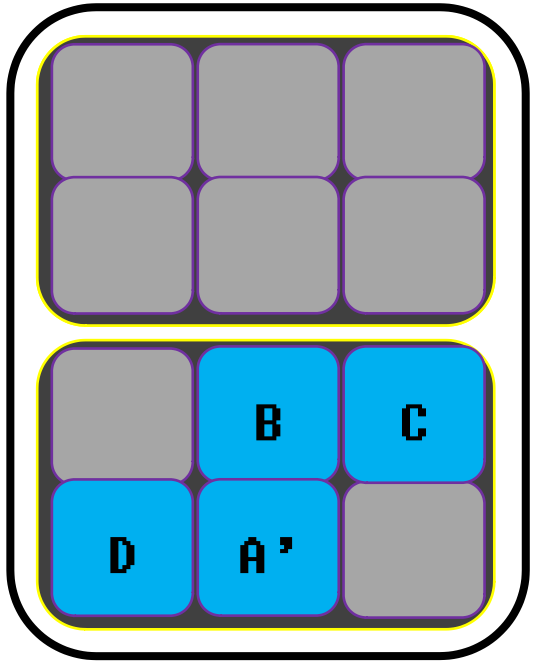
- Operações de:
  - Reescrita
  - Apagamento
  - Coleta de lixo

Disco de estado sólido (SSD)



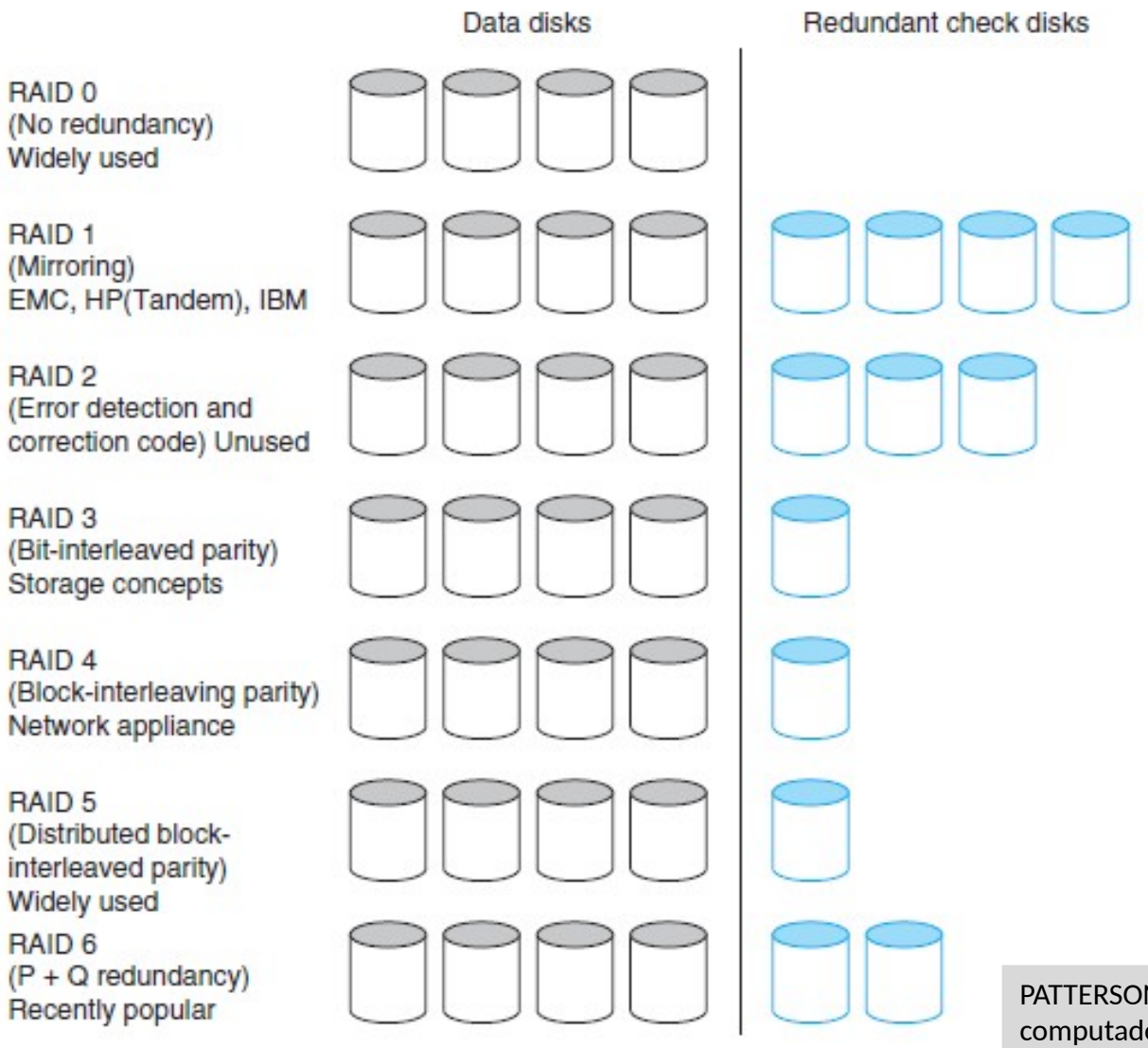
- Operações de:
  - Reescrita
  - Apagamento
  - Coleta de lixo

Disco de estado sólido (SSD)



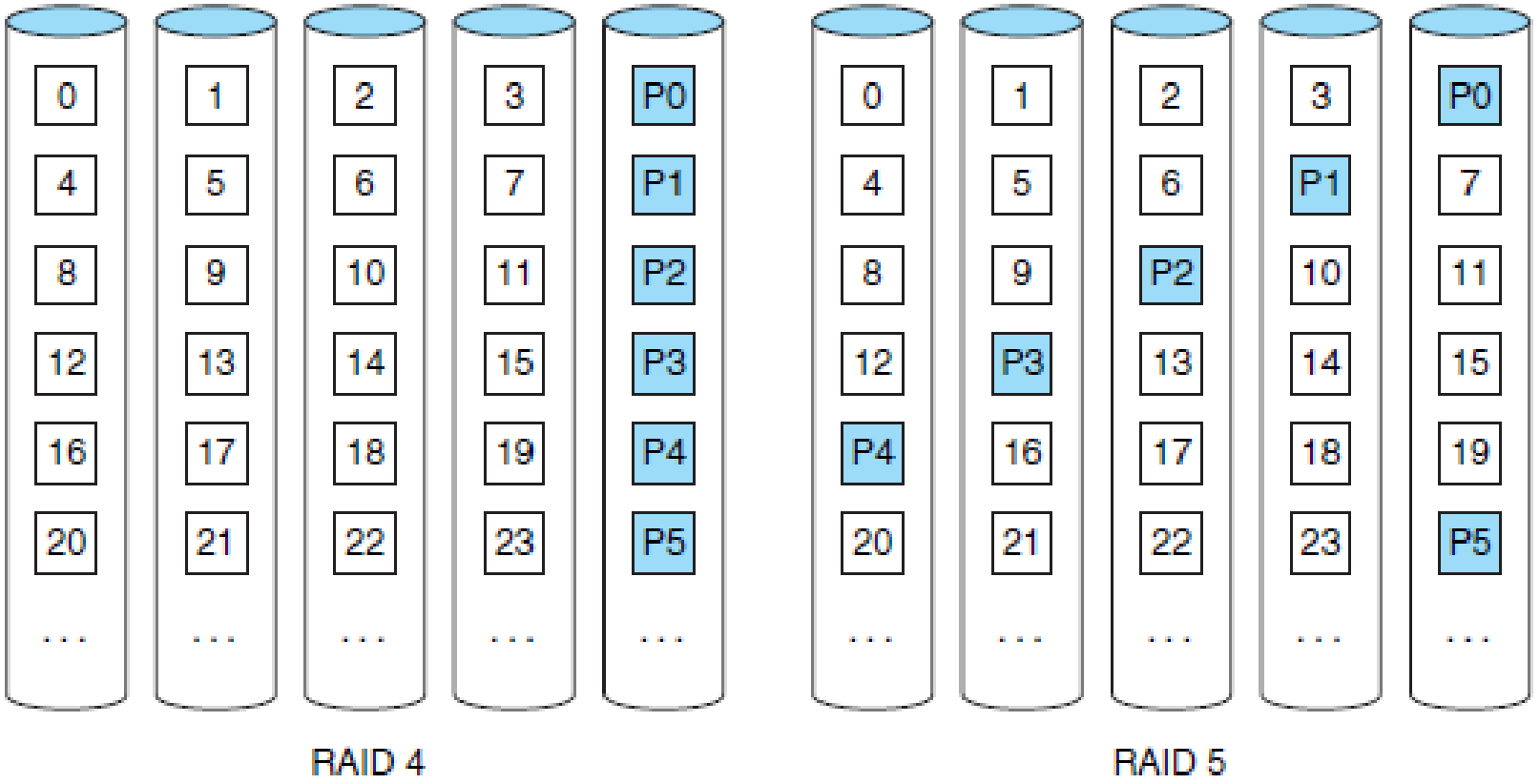
- Operações de:
  - Reescrita
  - Apagamento
  - Coleta de lixo

Redundant Arrays of Inexpensive Disks (RAID)



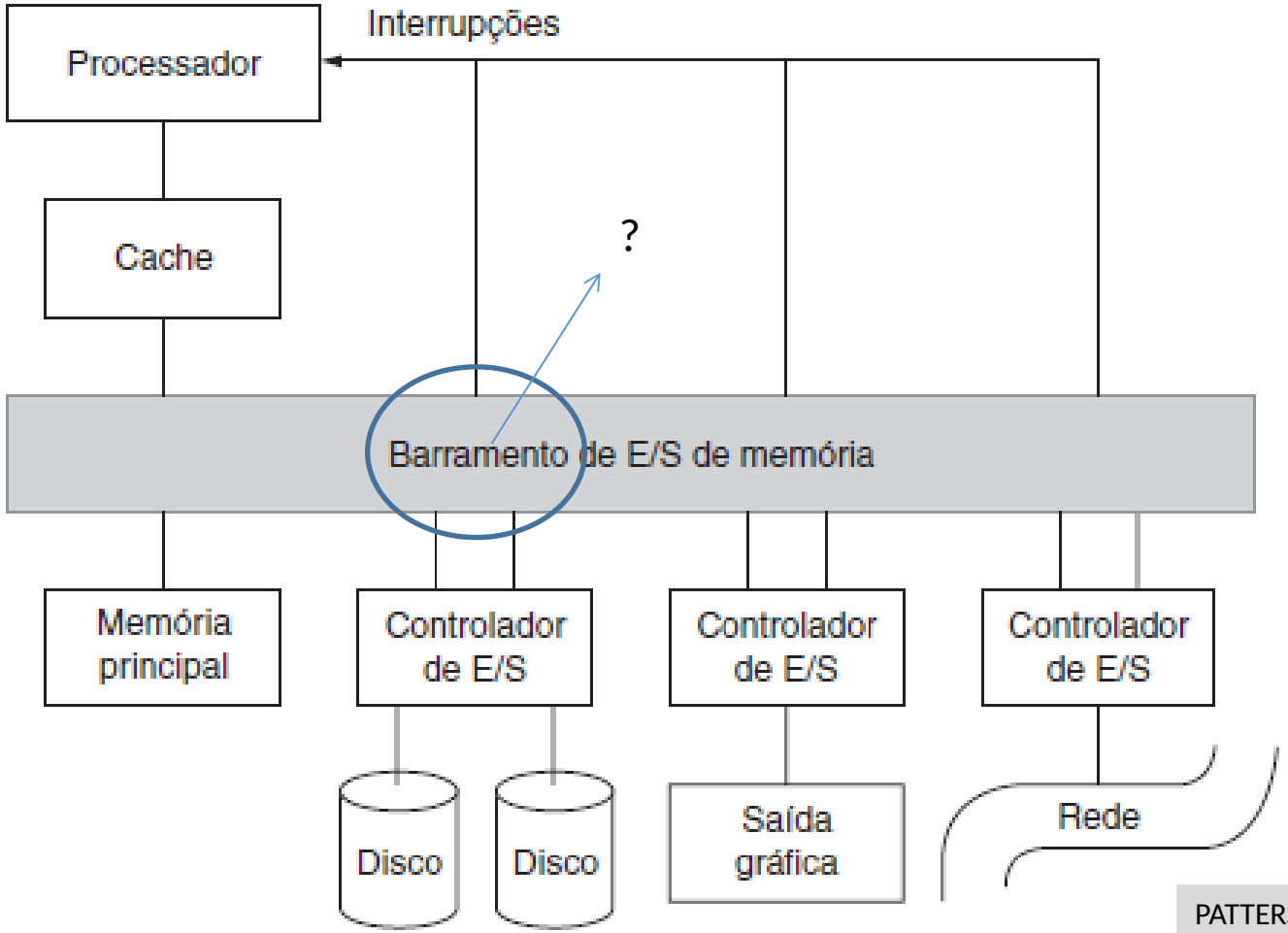
PATTERSON, David A. Organização e projeto de computadores a interface hardware/software

Redundant Arrays of Inexpensive Disks (RAID)



PATTERSON, David A. Organização e projeto de computadores a interface hardware/software

Visão geral



PATTERSON, David A. Organização e projeto de computadores a interface hardware/software