ANHANGUERA EDUCACIONAL

Sistemas Operacionais

Marcos Grillo

Literatura

- MACHADO, Francis Berenger; MAIA, Luiz Paulo (orgs.). **Arquitetura de Sistemas Operacionais**. 4ª ed. Rio de Janeiro: LTC - Livros Técnicos e Científicos, 2008

Programa Livro-Texto.

Conteúdo Programático
Conceitos básicos de sistemas operacionais, uma visão geral:
Sistemas Monoprogramáveis/Monotarefa,
Sistemas Multiprogramáveis/Multitarefa,
Sistemas com Múltiplos processadores,
Sistemas Fortemente acoplados,
Sistemas Fracamente acoplados.
Estrutura do Sistema Operacional
Processo:
Modelo de processo, estados, mudanças de estados,
Subprocesso e Thread,
Tipos de processos.
Comunicação entre processos
Especificação de concorrência em programas,
Problemas de compartilhamento de recursos,
Problemas de sincronização,
Deadlock.
Gerência do Processador:
Critérios de Escalonamento,
Escalonamento Não-preenptivo,
Escalonamento Preenptivo,
Escalonamento com Múltiplos Processadores
Gerência de Memória:
Alocação Contígua Simples,
Alocação Particionada,
Memória Virtual,
Segmentação, segmentação com paginação,
Proteção,
Compartilhamento de memória.

Sistema de Arquivos:
Organização de Arquivos,
Métodos de acesso, operações de I/O e Atributos,
Diretórios,
Alocação de espaço em disco,
Proteção de acesso,
Implementação de Cachês.
Gerência de Dispositivos:
Operações de I/O,
Subsistemas de I/O,
Device Drivers,
Controladores,
Dispositivos de Entrada/Saída

Ementa – 1ª etapa.

- Introdução a sistemas operacionais;
- Visão geral de sistemas operacionais;
- Conceitos básicos de SO, hardware e software; Conceitos básicos de SO, hardware e software;
- Estrutura do Sistema Operacional;
- Tipos de processos e Processos e Threads;
- Processos e Threads;
- Sincronização e comunicação entre processos e threads;
- Revisão, exercícios, seminários;

Ementa - 2ª etapa.

- Gerência do processador;
- Gerência de memória;
- Gerência de dispositivos;
- Sistemas com múltiplos processadores;
- Sistemas operacionais comerciais/Livre;
- Prova escrita oficial;
- Revisão;
- Prova Substitutiva;

Horários.

- 1ª aula 19:10 – 20:00
- 2ª aula 20:00 – 20:50
- 3ª aula 21:10 – 22:00
- 4ª aula 22:00 – 22:50 – Orientação ATPS

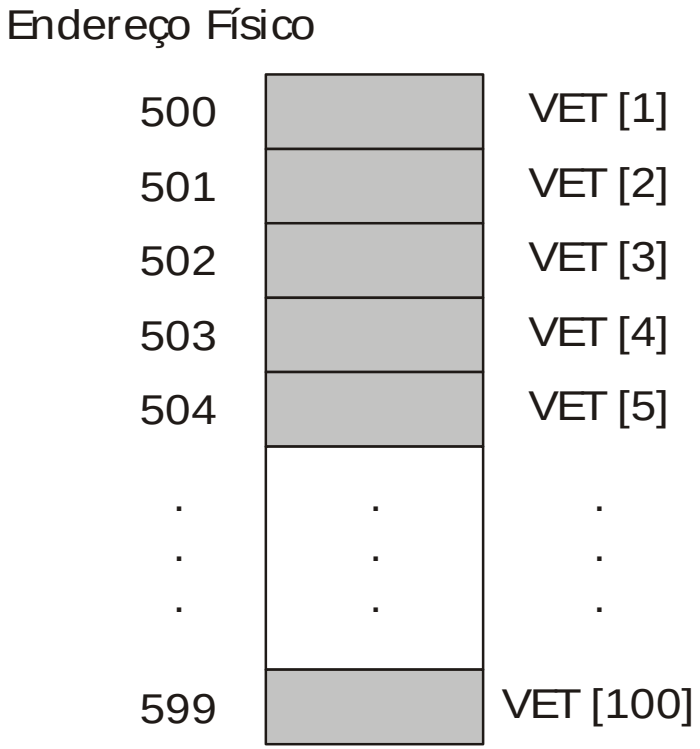
Avaliação.

- 1º Bimestre peso 4;
  - Prova + ATPS
- 2º Bimestre peso 6;
  - Prova + ATPS

Gerência de memória virtual.

Gerência de memória virtual consiste na técnica de combinar a memória real com a memória secundária, dando ao usuário a ilusão de existir uma memória muito maior que a capacidade real da memória principal.

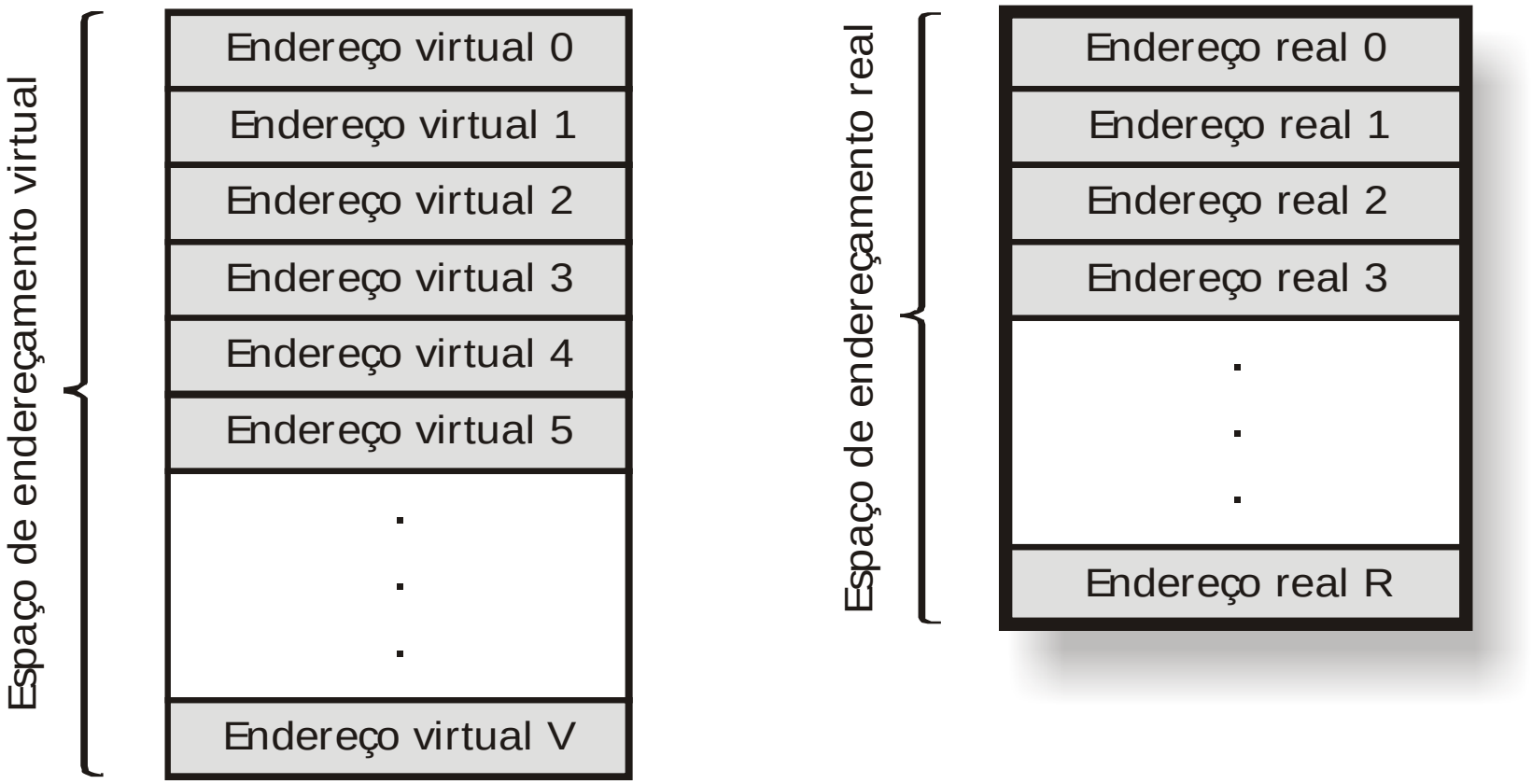
Espaço de endereçamento virtual.



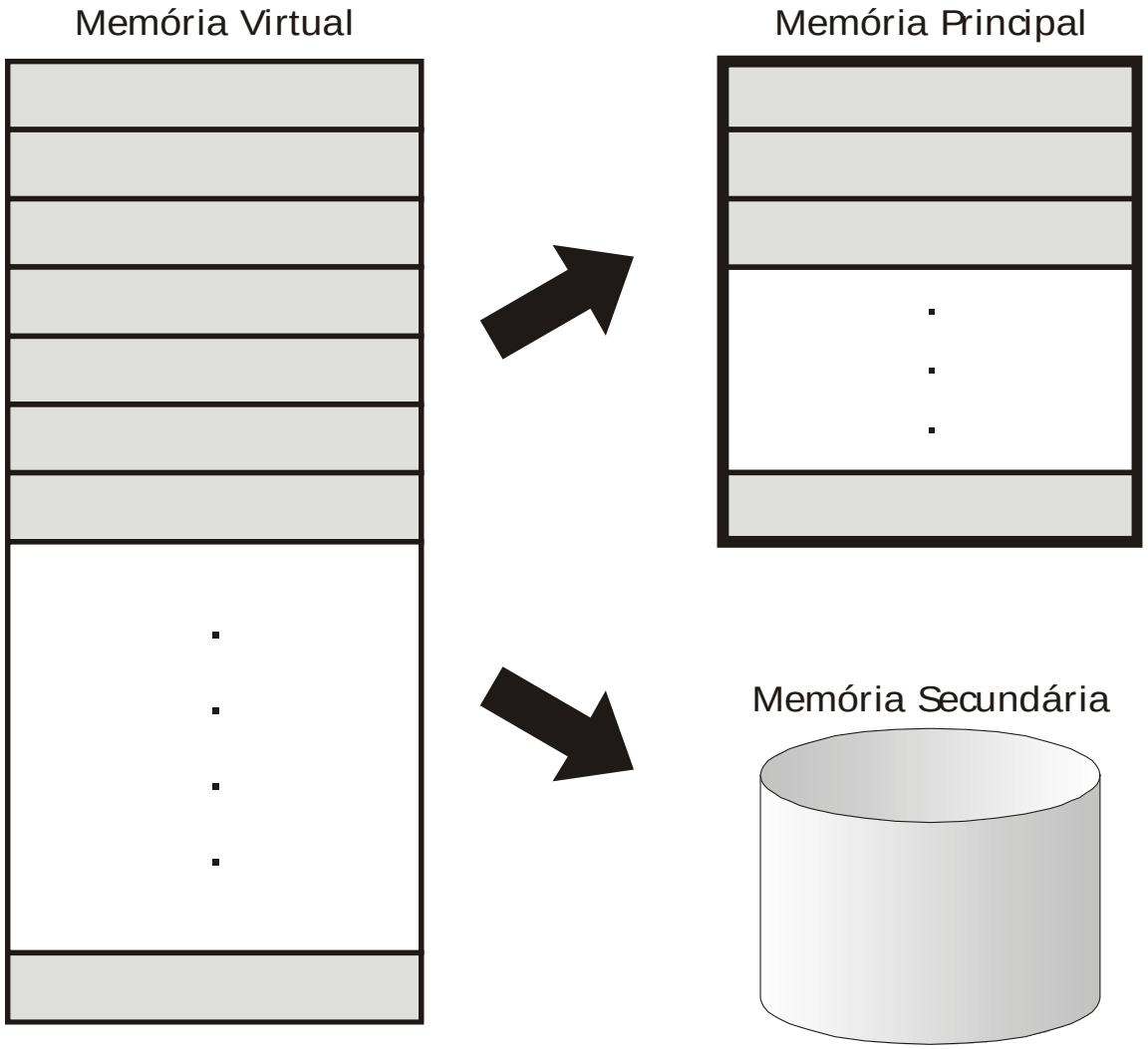
Espaço de endereçamento virtual.

- Como um vetor, o processo não precisa saber a posição da memória que se localiza um dado;
- Mecanismo de tradução do endereço virtual é denominado de mapeamento;
- O programa não precisa contar apenas com o espaço total da memória física.
- Apenas uma parte do programa fica residente na memória principal;
- Compiladores e Linkers geram o código executável em função aos endereços de memória virtual.

Espaço de endereçamento virtual.



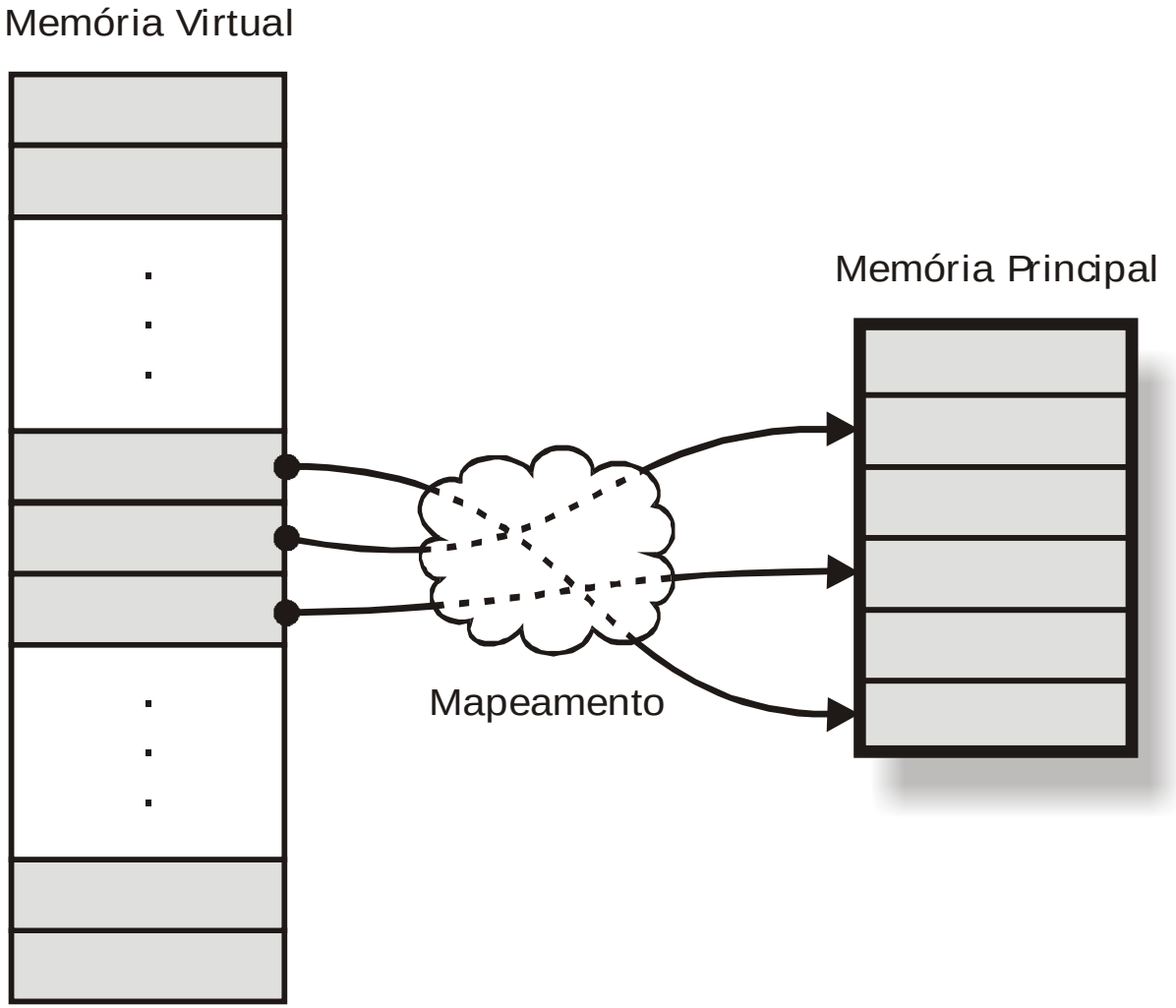
Espaço de endereçamento virtual.



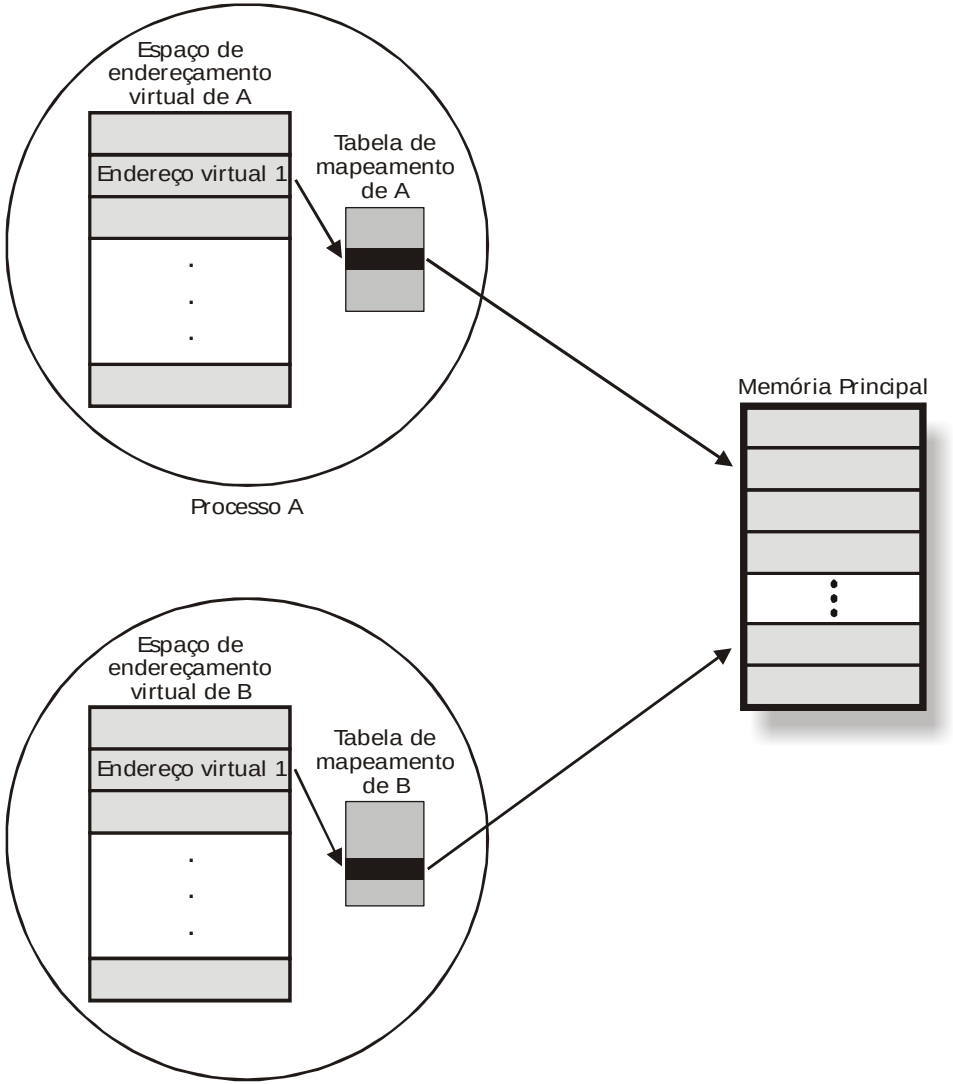
Mapeamento

- Processador executam apenas funções residentes na memória real;
- Sistemas modernos utilizam MMU para mapear (Hardware dedicado);
- Tamanho fixo (Técnica de paginação);
- Tamanho variável (Técnica de segmentação);
- Mista, fixo + variável.

Mapeamento



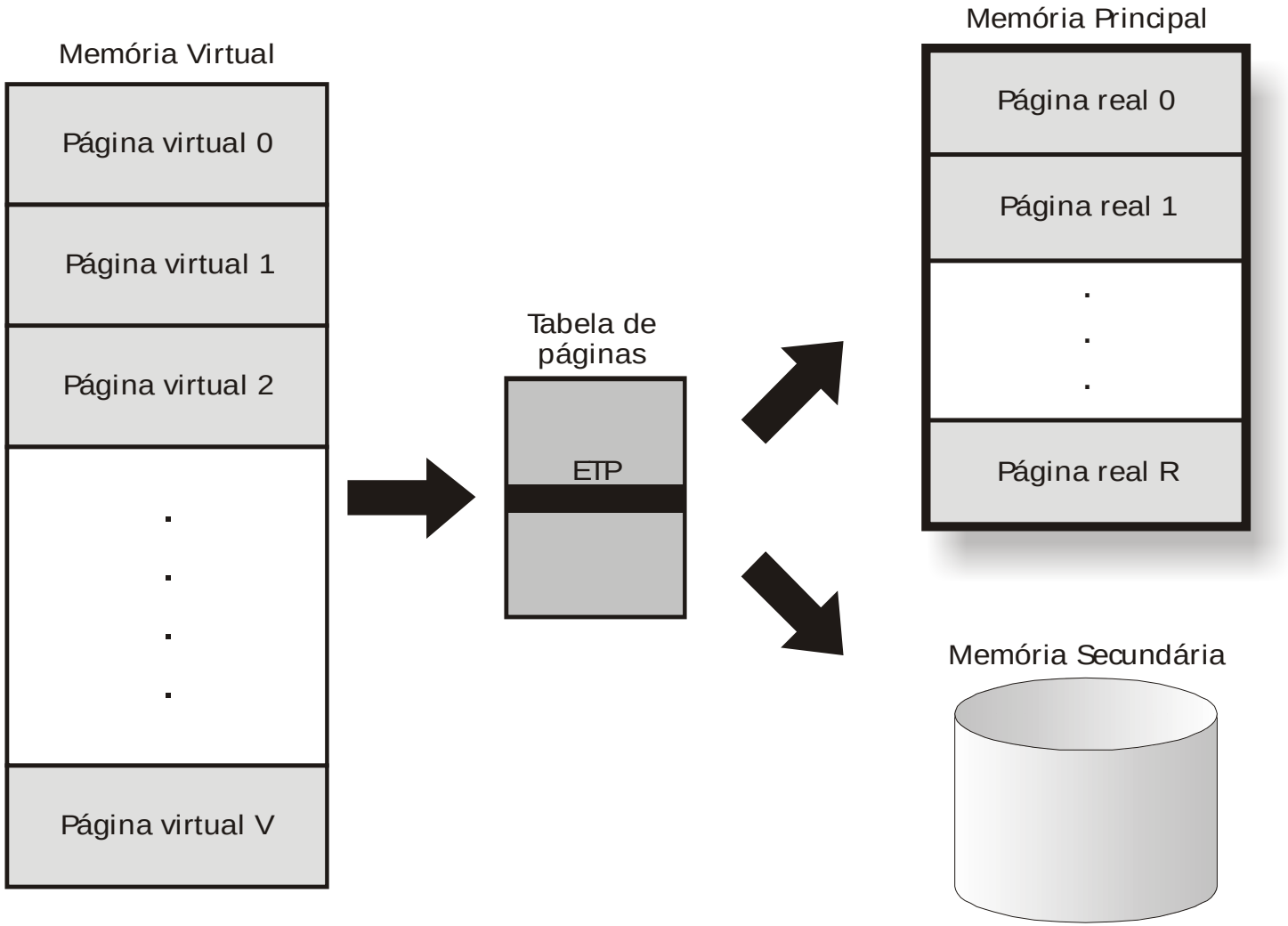
Mapeamento.


Espaço de endereçamento virtual de A
Endereço virtual 1
Tabela de mapeamento de A
Processo A
Memória Principal
Espaço de endereçamento virtual de B
Endereço virtual 1
Tabela de mapeamento de B
Processo B

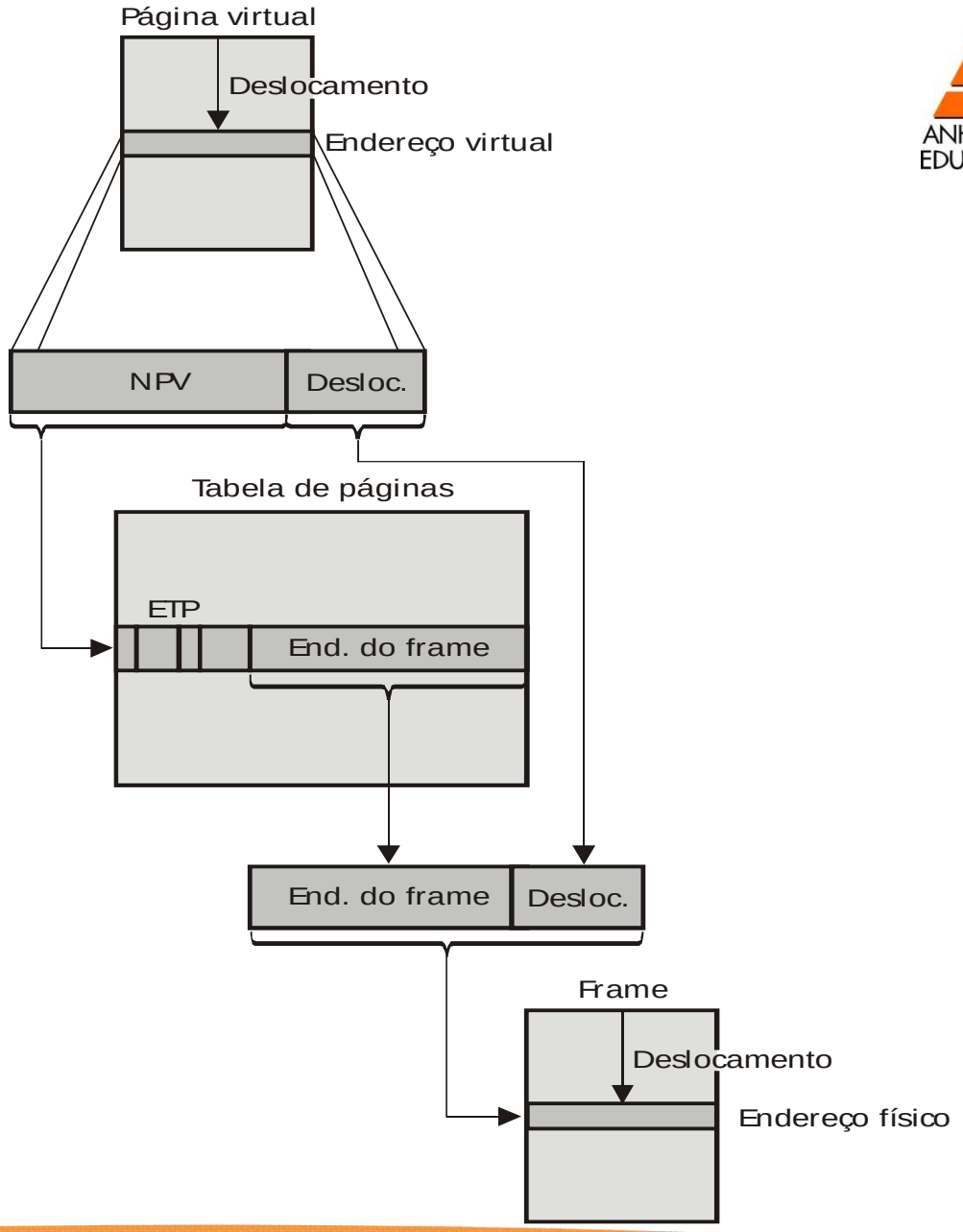
Memória virtual por paginação

- Blocos do mesmo tamanho (Paginas);
- *Page fault;*
  - Quando um processo referencia um endereço que não está na memória principal, ocorre um page fault.
- Utiliza a tabela de paginas;
- NPV + Deslocamento = endereço do dado;
- NPV (número da pagina virtual) – Índice da tabela de paginas;
- ETP(entrada da tabela de páginas):
  - *Valid bit* (0 não localizada na memória principal e 1 localizada).

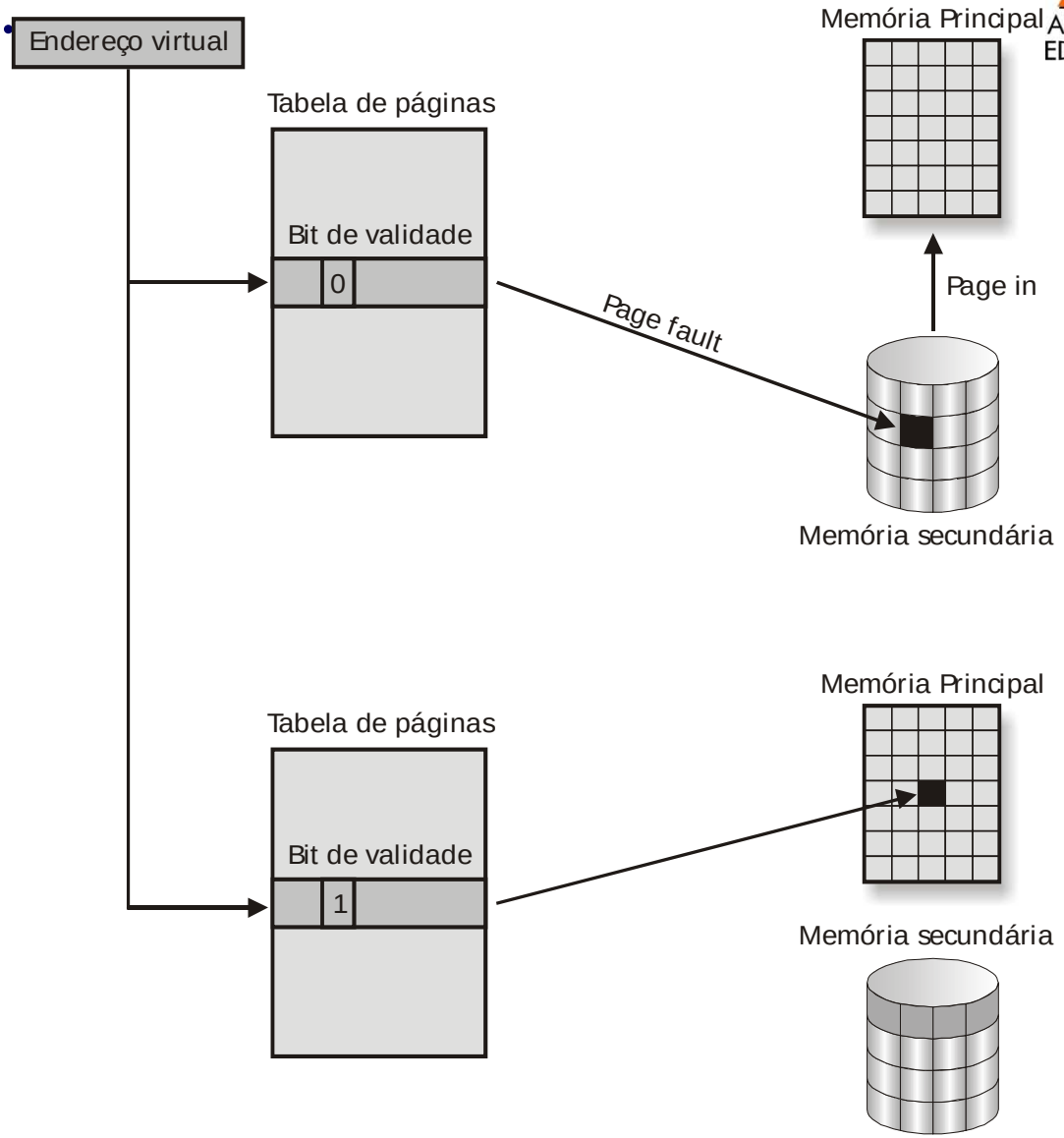
Memória virtual por paginação.



Memória virtual por paginação.



Memória virtual por paginação.



Políticas de busca de páginas.

- Paginação por demanda (*demand paging);*
  - Transferida apenas quando referenciadas.
- Paginação antecipada (*antecipatory paging / prepaging*);
  - Carrega a página referenciada e outras que podem ser utilizadas.
  - Pode ocorrer no momento da criação de um processo ou em um *page fault*.

Políticas de Alocação de Páginas.

ANHANGUERA EDUCACIONAL

- Determina quantos frames cada processo pode manter na memória principal;
- Politica de locação fixa;
  - Definida no momento de criação de um processo (contexto de software);
  - Problemas:
    - Número de páginas pequeno maior *page fault*;
    - Número de páginas grande, menor números de processos em memória.
- Politica de locação variável;
  - O número de páginas podem variar durante a execução do processo.
  - Quanto mais *page faults*, mais dados são alocados na memória principal.

Políticas de Substituição de Páginas.

Em algumas situações, quando um processo atinge seu limite de locação de frames e necessita alocar mais páginas na memória principal, o sistema operacional deve selecionar quais páginas devem ser liberadas.

Livro texto, página 183

Políticas de Substituição de Páginas.

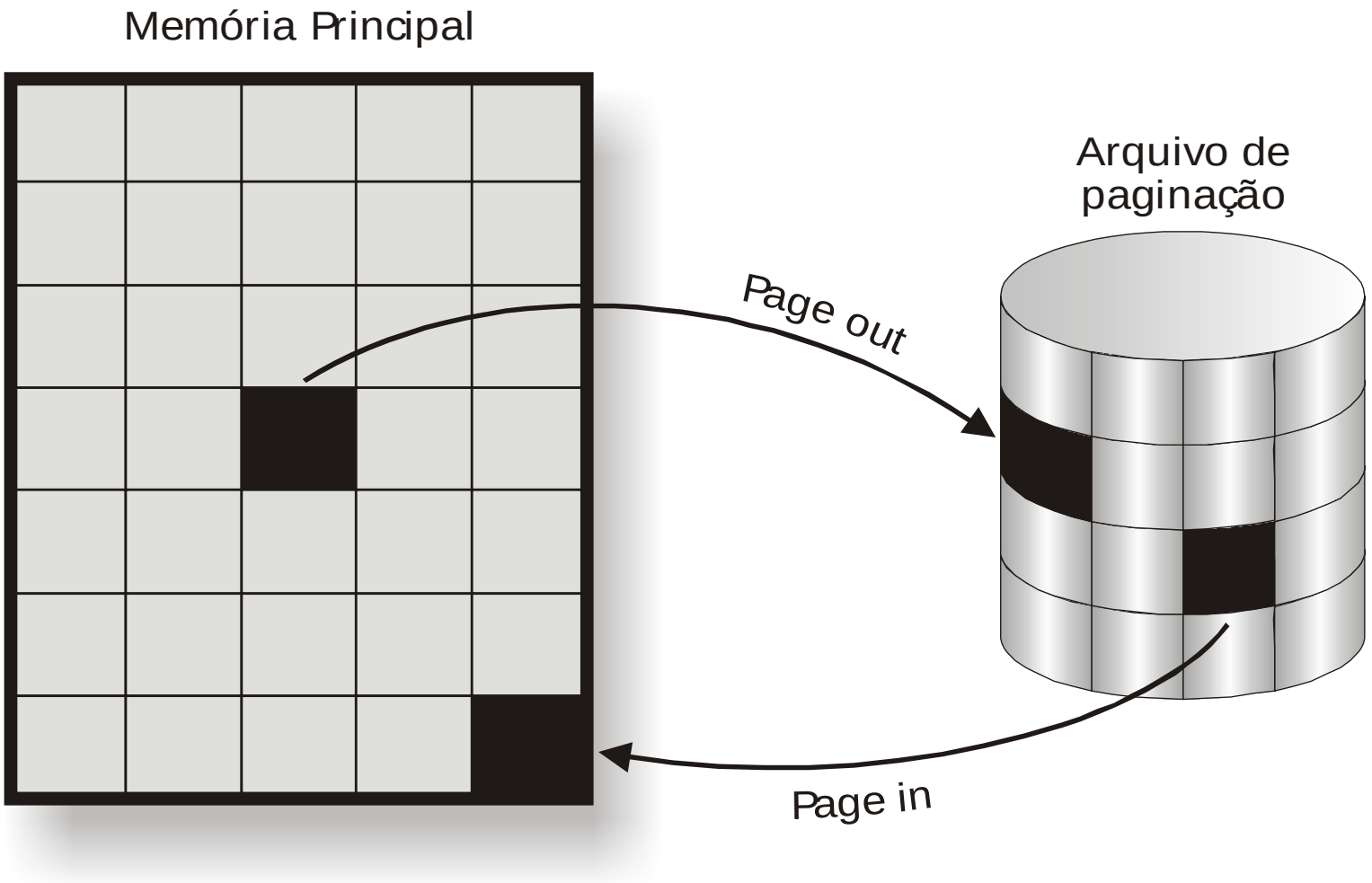
- O Sistema Operacional deve verificar:
  - A página foi ou não modificada, evitar perdas de dados;
  - Se modificada gravar antes de liberar na memória secundária;
  - Mecanismo com bit de verificação de alterações (dirty bit / modify bit).
  - Bit de verificação localizado na tabela de páginas.
- Executáveis não sofrem alterações, grandes candidatos a deixarem a memória principal;
- Arquivo de paginação (*page file*);

Políticas de Substituição de Páginas.

Entre páginas da memória principal:

- Política de substituição local:
  - Apenas páginas do processo.
- Política de substituição global:
  - Páginas de qualquer processo;
  - Páginas do núcleo do sistema não podem ser realocadas, marcadas como bloqueadas;
- Política de locação variável:
  - Tanto local quanto global;
  - Aumenta ou diminui o número de paginas para o processo.

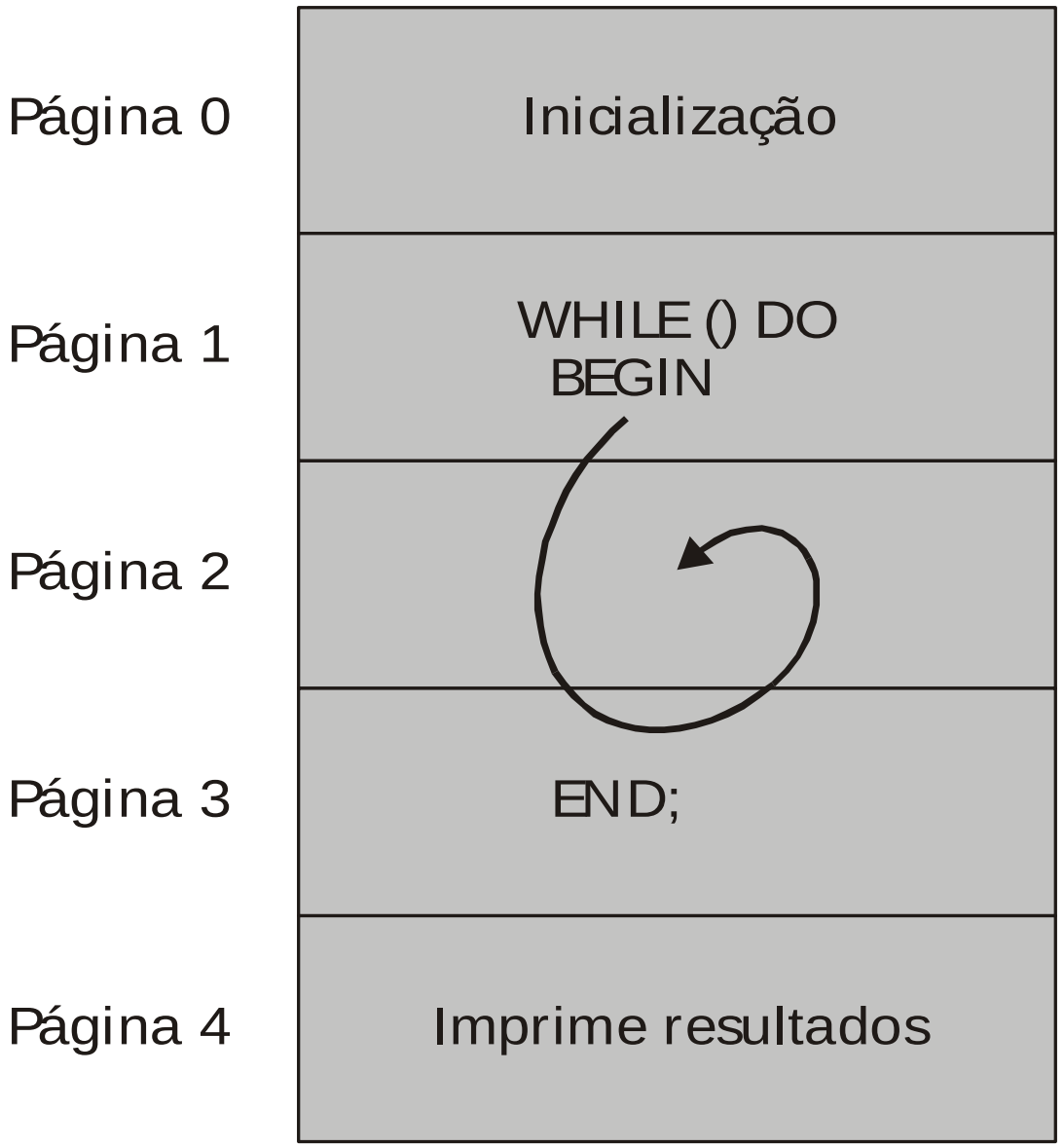
Políticas de Substituição de Páginas..



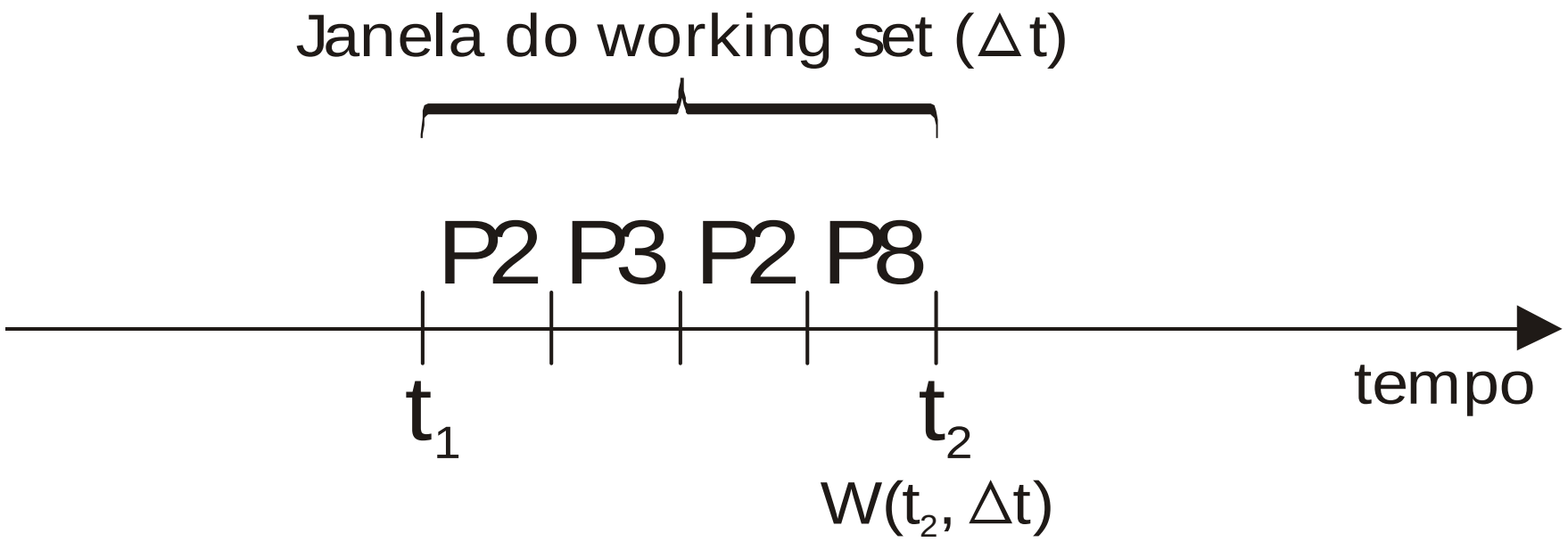
Working Set.

- Observa-se o número elevado de *page faults;*
- Com um número elevado de *IO* ocorre o problema conhecido como *thrashing;*
- Busca resolver o problema de desempenho;
- Utiliza o princípio de localidade, ex. loop;

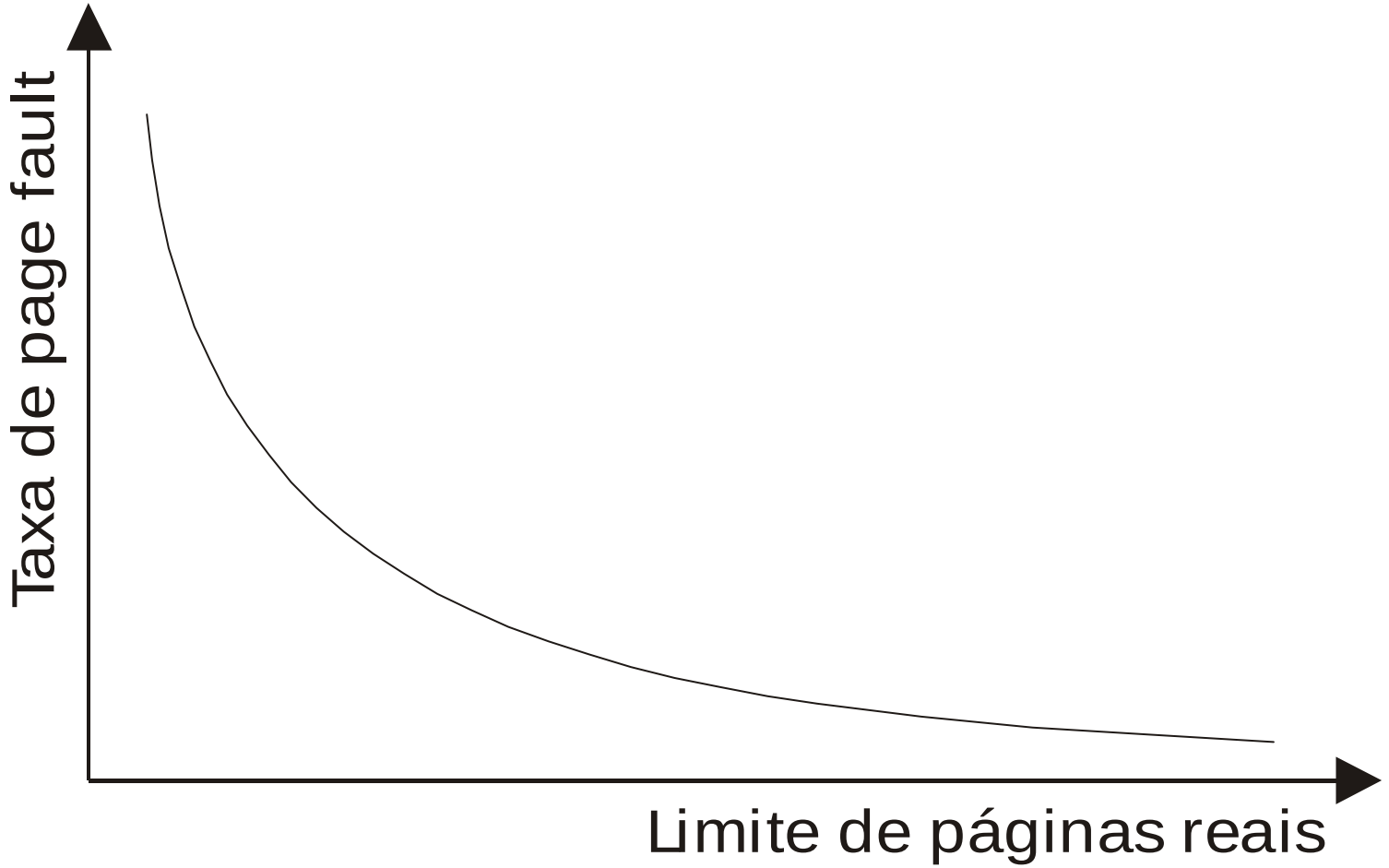
Working Set.


Página 0
Página 1
Página 2
Página 3
Página 4
Inicialização
WHILE () DO
BEGIN
END;
Imprime resultados

Working Set.



Working Set.

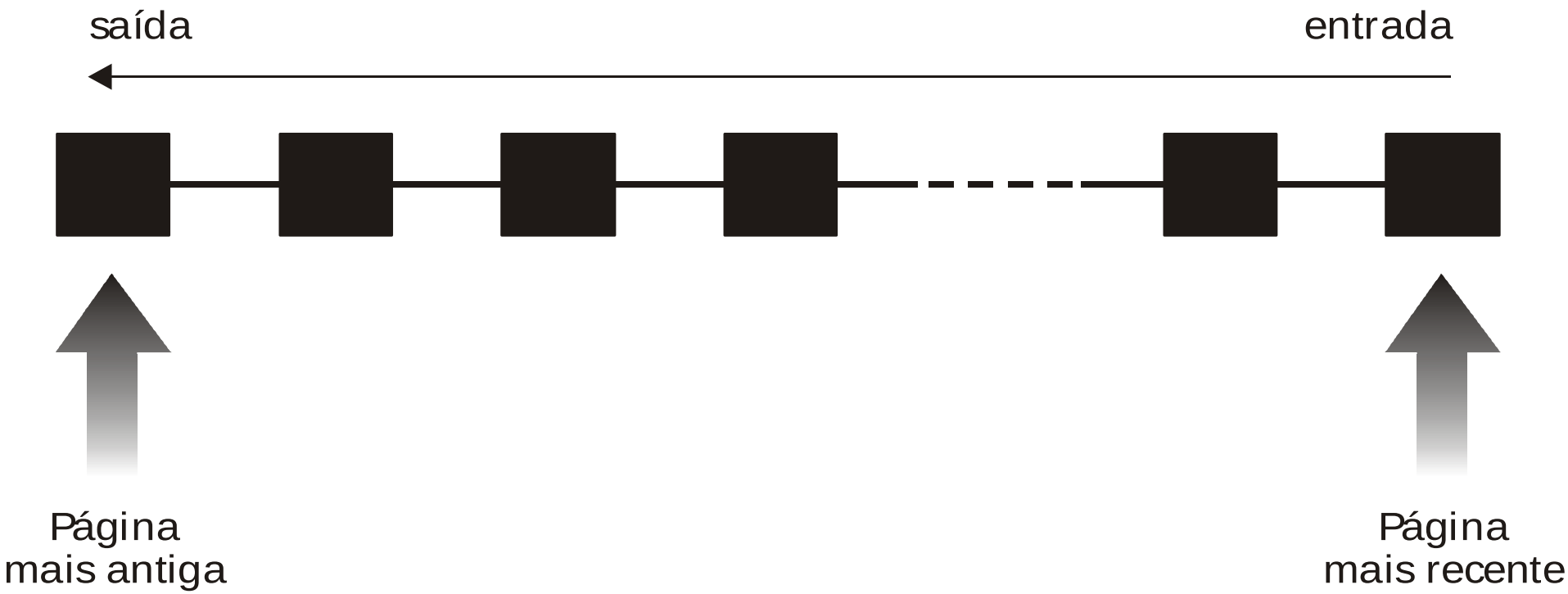


Algoritmos de substituição de páginas.

Grande problema é quem substituir e não quem carregar... E quem precisa ser substituído?

- Menores chances de serem referenciados;
- Algoritmo mais sofisticado maior *overhead;*
- Entre os algoritmos:
  - Ótimo;
  - Aleatório;
  - FIFO;
  - LFU (Least-frequently-Used);
  - LRU (Least-recently-Used);
  - NRU (Not-recently-Used).

FIFO.



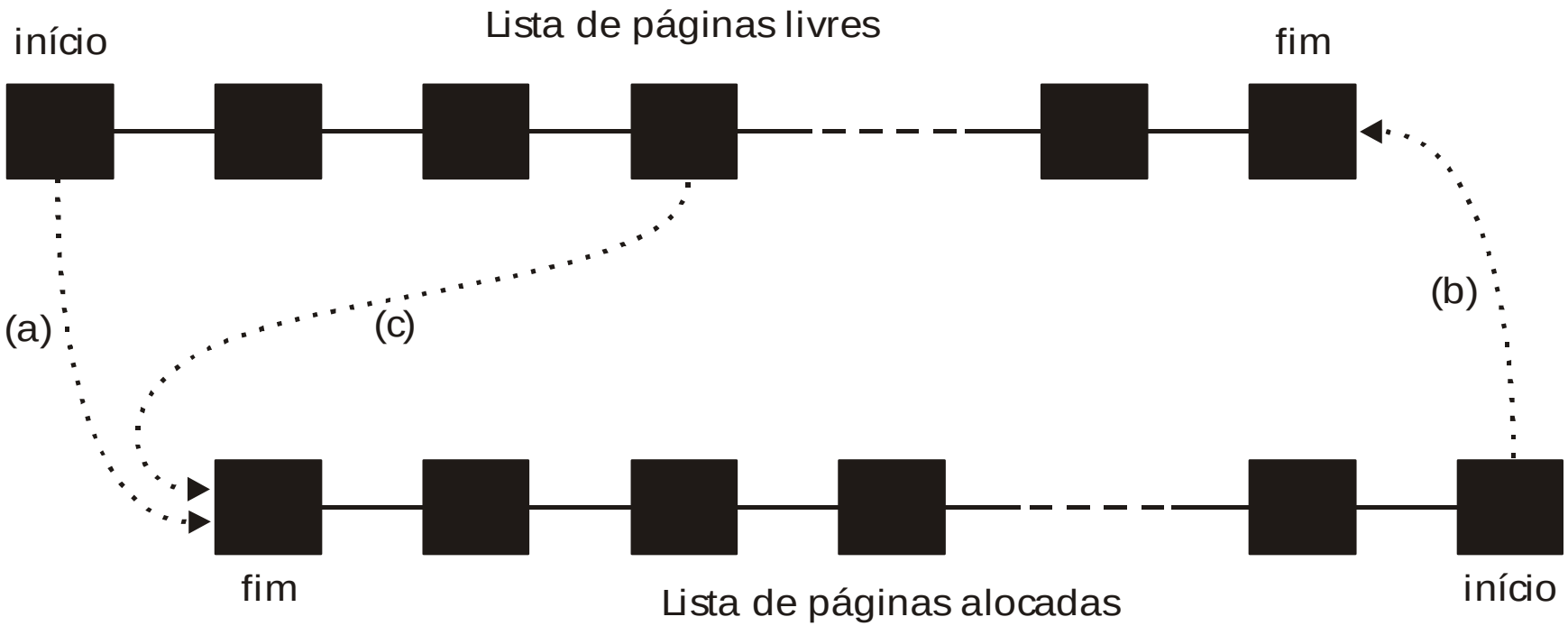
Problemas dos algoritmos.

- Ótimo;
  - Impossível de ser implementado, SO não consegue reconhecer o comportamento futuro dos processos.
- Aleatório;
  - Retira qualquer página, não reconhece a frequência de utilização.
- FIFO.
  - Segue uma fila uniforme, retirando o frame com mais tempo de permanência na memória.

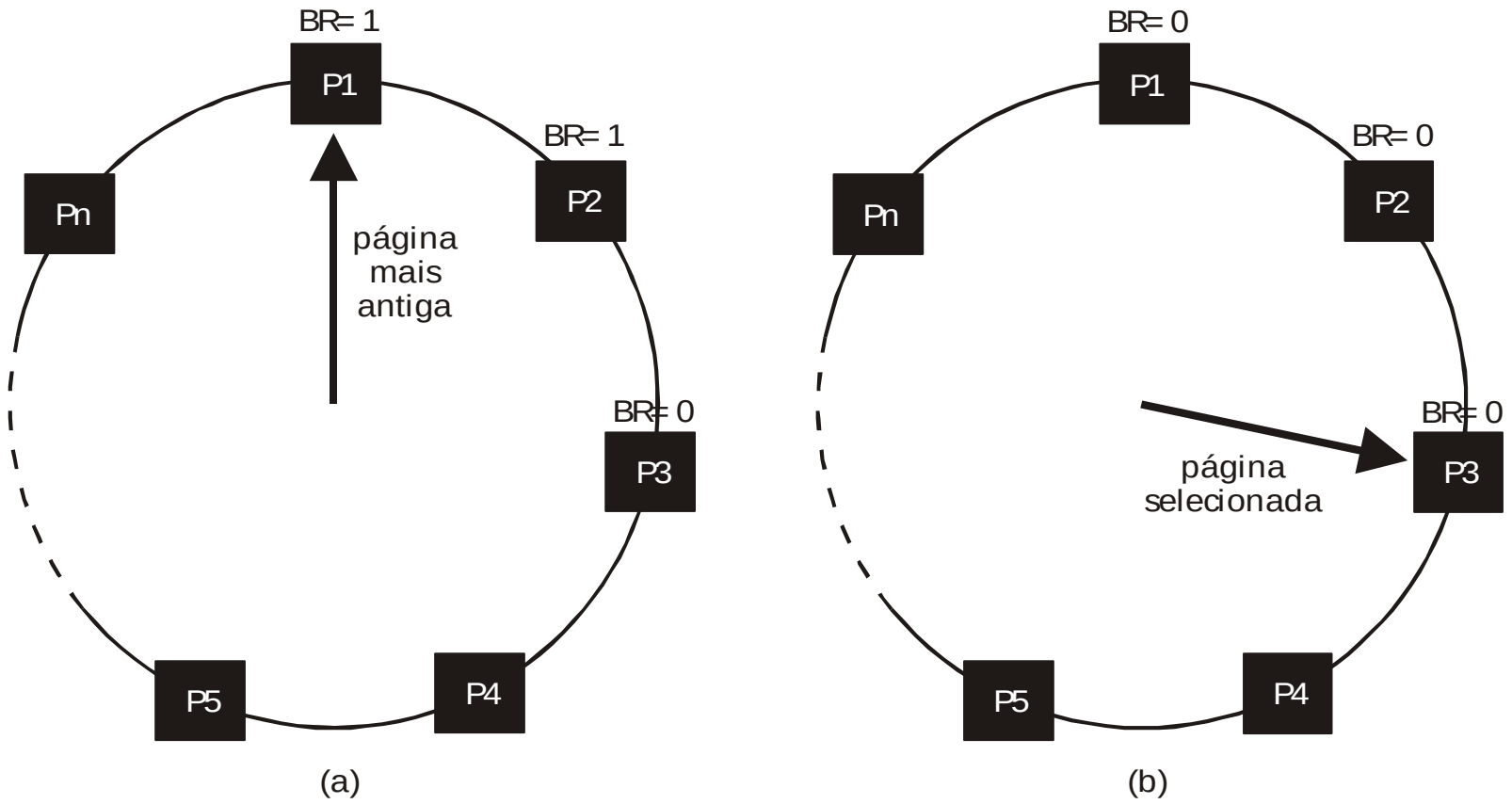
Problemas dos algoritmos.

- LFU (Least-frequently-Used);
  - Páginas com pouco tempo de memória podem ser retiradas, por não conter um histórico.
- LRU (Least-recently-Used);
  - Elevado custo de implementação, consumo de recursos.
- NRU (Not-recently-Used).
  - Precisa ser utilizada com dois bits, o de referência e o de modificação, elevado custo de implementação.

FIFO com buffer de páginas.



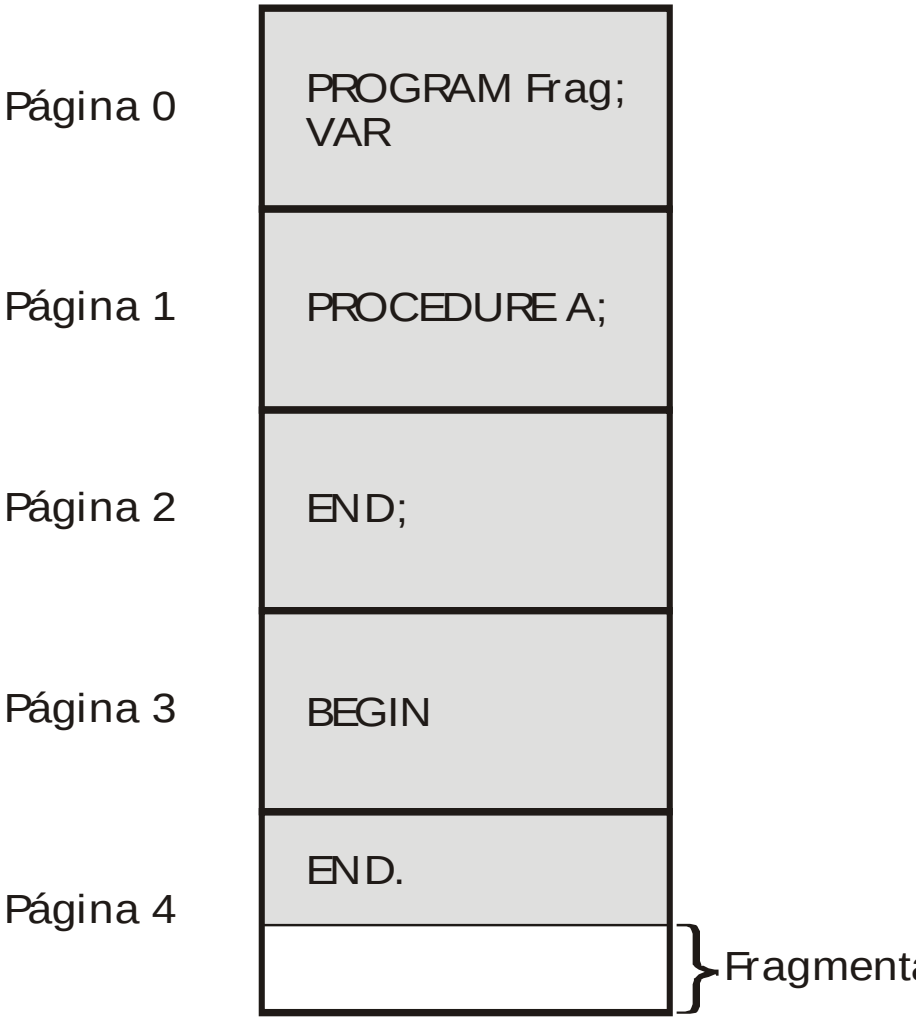
FIFO com buffer de páginas.



BR=Bit de Referência

Tamanho de páginas

- Menor o tamanho de página menor fragmentação;
- Maior o tamanho da página por gerar espaços maiores de fragmentação;



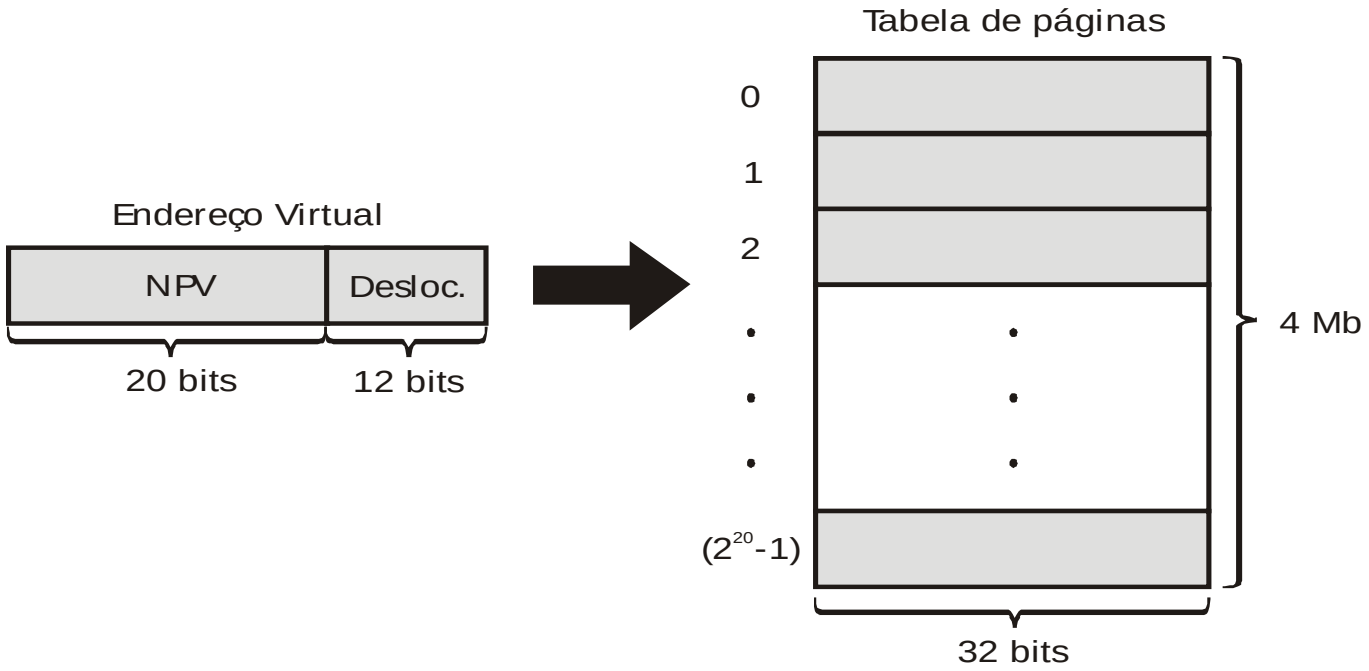
Tamanho de páginas

- É mais demorado a operação de *IO* de duas paginas de 512 bytes que uma página de 1024 bytes;

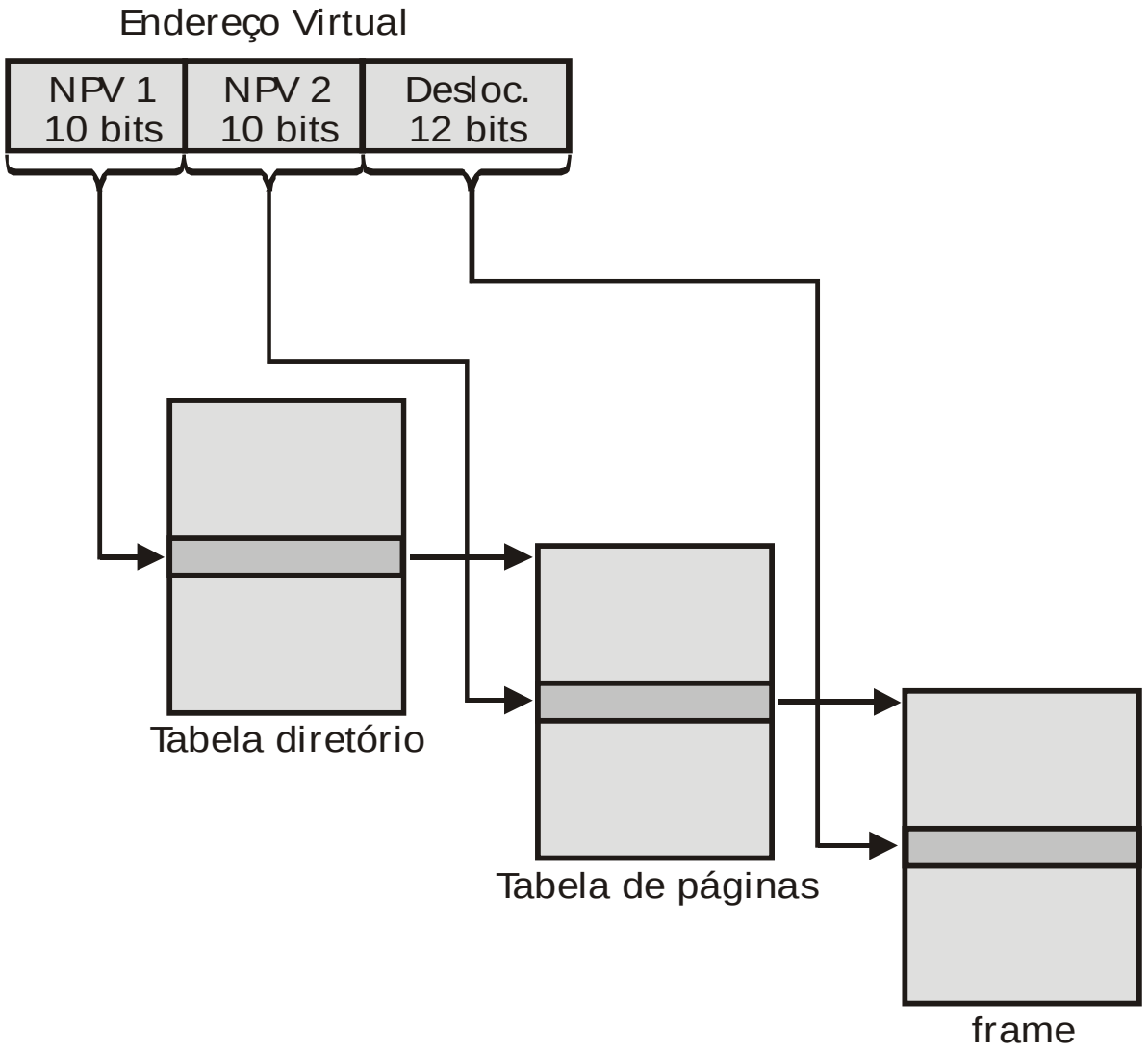
- Tendência de aumento do tamanho das páginas em sistemas mais modernos, devido aos recursos secundários melhorarem.

Paginação em Múltiplos Níveis.

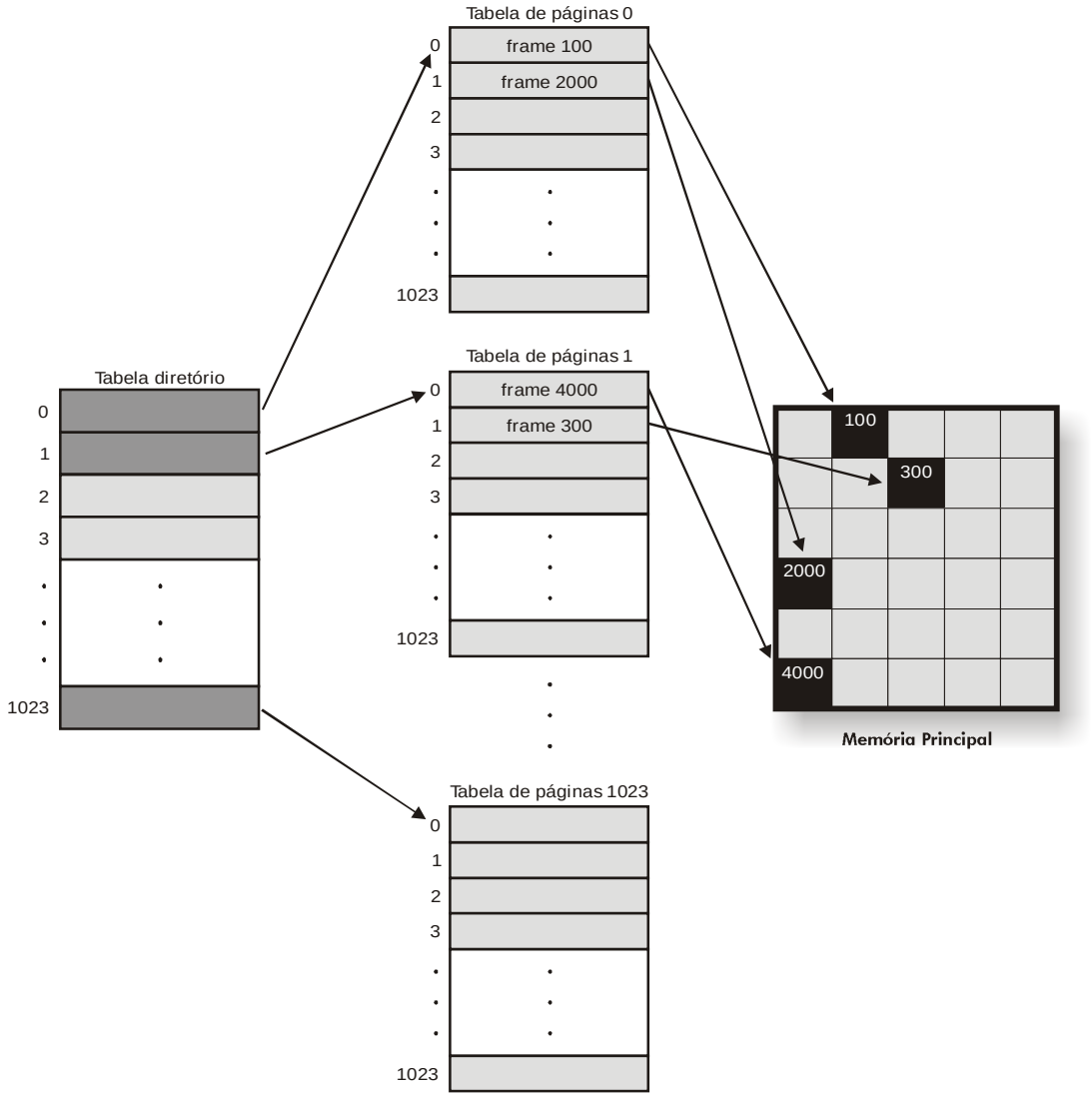
- Tabela de páginas consumindo muito espaço;
- Apenas informações realmente necessárias aos processos estariam na tabela principal;



Paginação em Múltiplos Níveis.



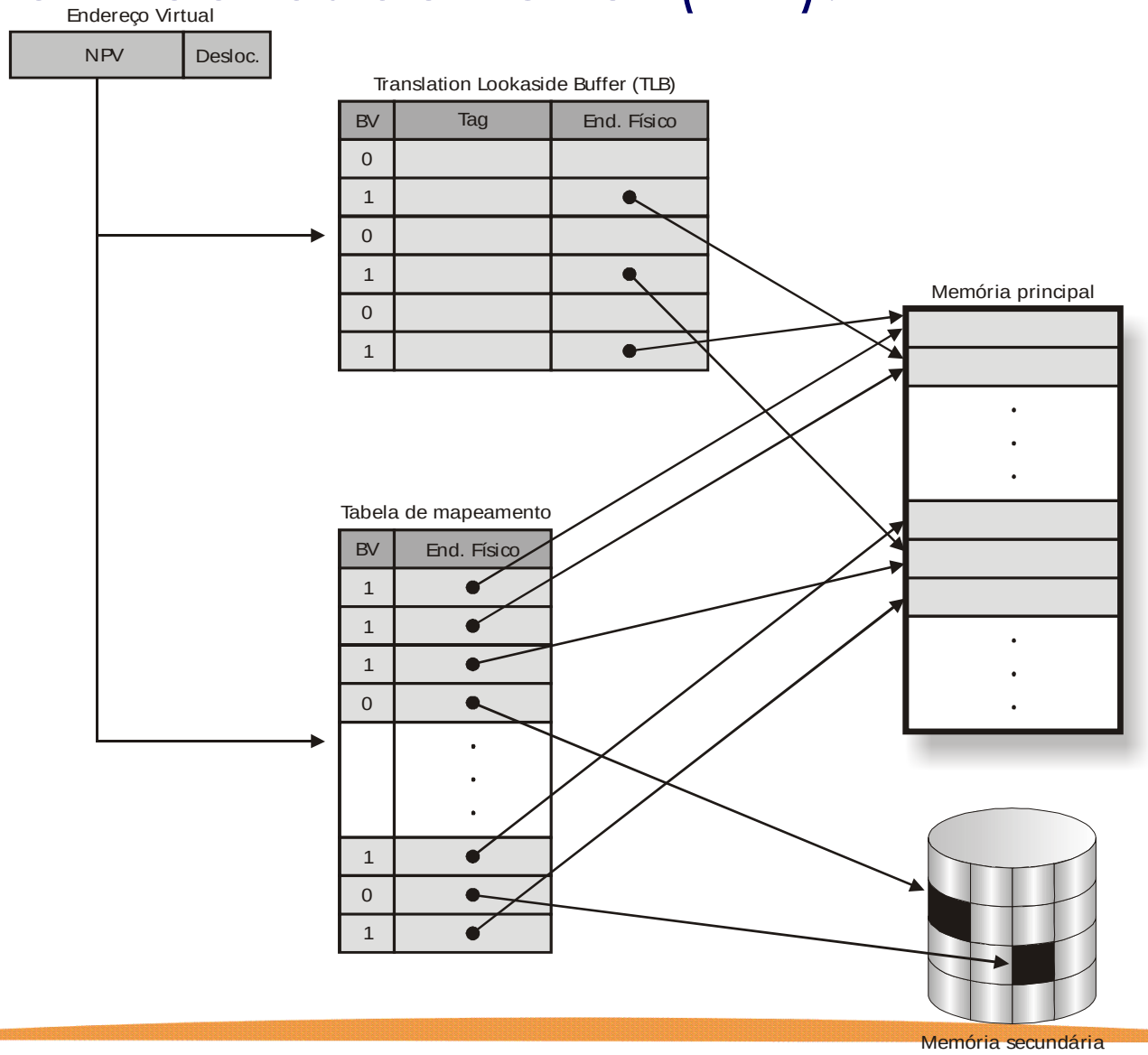
Paginação em dois níveis.



Translation Lookaside Buffer (TLB)

- Poucas tabelas são referenciadas com maior frequência;
- TLB é como uma memória cache;
- Evita acesso excessivo na tabela de páginas;
- Utiliza mapeamento associativo (não precisando percorrer toda TLB para encontrar o endereço.

Translation Lookaside Buffer (TLB).



Translation Lookaside Buffer (TLB).

Campo	*Descrição*
Tag	Endereço virtual sem deslocamento
Modificação	Bit que indica se a página foi alterada
Referência	Bit que indica se a página foi recentemente referenciada, sendo utilizada para realocação de entrada pela TLB
Proteção	Define a permissão de acesso a página
Endereço físico	Posição do frame na memória principal

Proteção de memória.

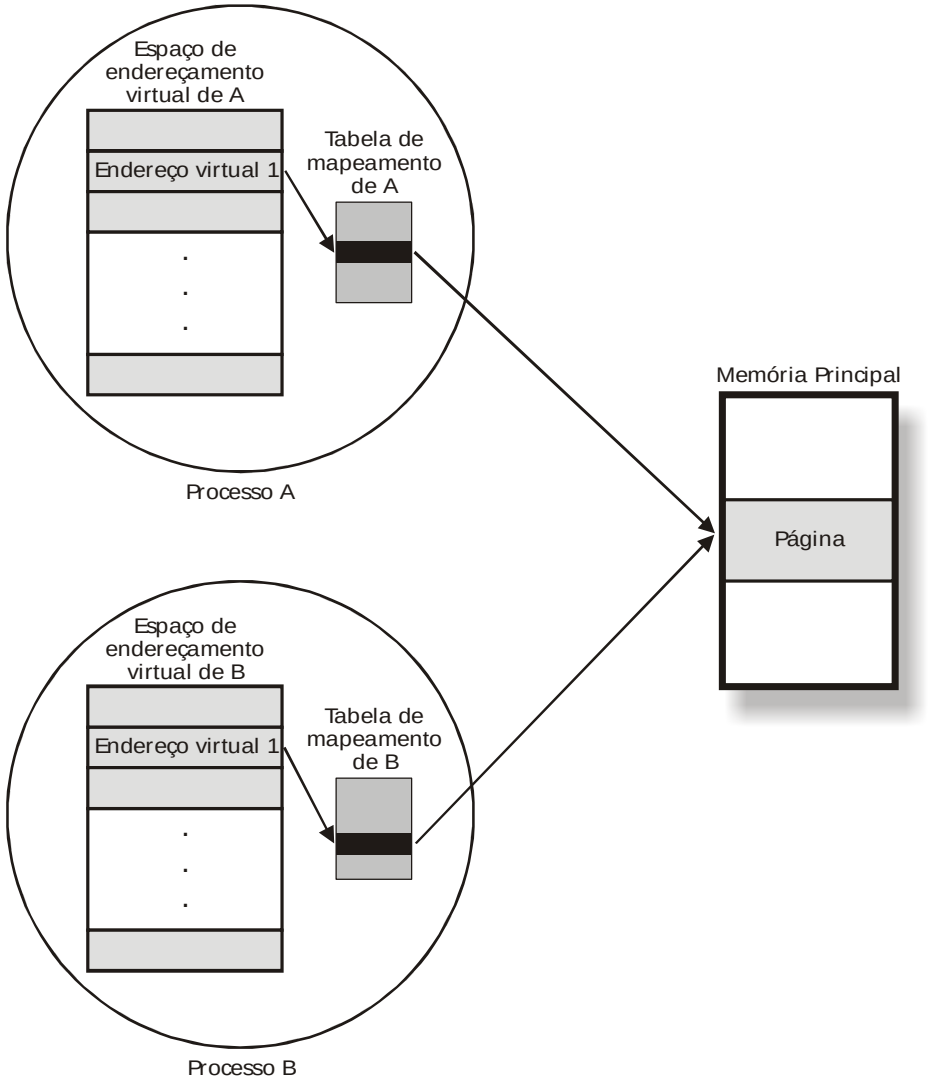
Importância:

- Vários programas acessando partes em comum da memória;
- Não alterar nenhuma informação localizada em um setor privilegiado (SO);
- Proteger informações do SO.

Compartilhamento de memória.

- Reentrância;
- Cada processo tem sua tabela de memória mas referenciam o mesmo frame;
- Aplicações que compartilham os mesmos dados da memória principal.

Compartilhamento de memória.



Memória virtual por segmentação

É a técnica de memória virtual que divide os blocos em tamanhos diferentes, chamados de segmentos.

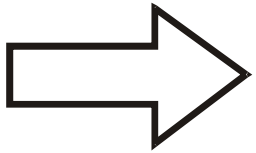
Os programas são divididos logicamente em sub-rotinas e estrutura de dados, que por sua vez serão alocados na memória principal.

Memória virtual por segmentação

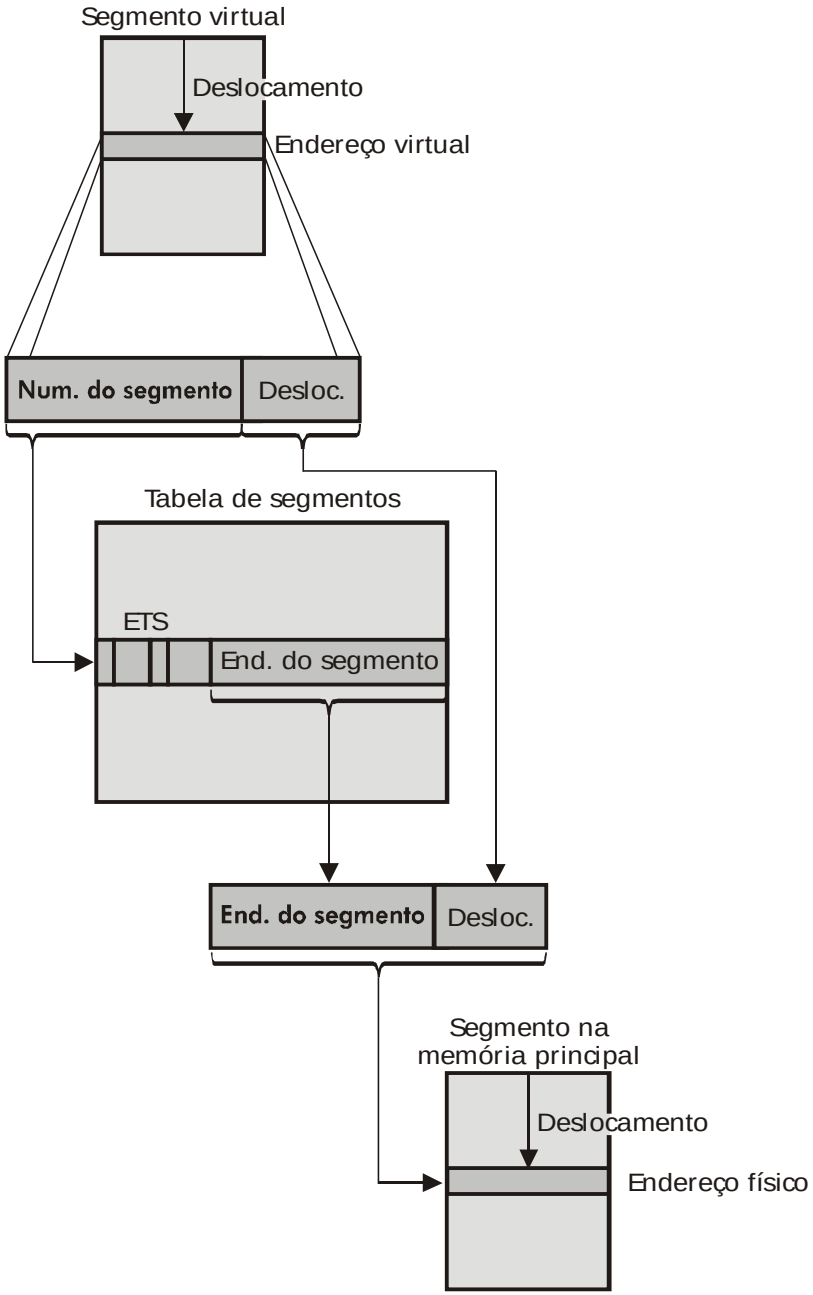
- Normalmente definido pelo compilador;
- Possui um numero máximo de segmentos;
- Utilizam tabelas de mapeamento de segmentos (TMS) ao invés de tabelas de paginação;
- Possuem numero de segmento virtual (NSV);
- Possuem deslocamento;
- Endereço é obtido utilizando TMS + NSV + Deslocamento;
- Fragmentação externa.

Memória virtual por segmentação

```
PROGRAM Segmento;
    VAR A: ARRAY...
    C: ...
PROCEDURE X;
END;
FUNCTION Y;

END;

BEGIN

END.
```



Memória virtual por segmentação.



Memória virtual por segmentação – tabela ETS

Campo	*Descrição*
Tamanho	Especifica o tamanho do segmento
Bit de validade	Indica se o segmento está na memória principal
Bit de Referência	Bit que indica se o segmento foi recentemente referenciado, sendo utilizado pelo algoritmo de substituição
Proteção	Define a permissão de acesso a página
Bit de modificação	Indica se o segmento foi alterado

Swapping em memória virtual.

- Não temos espaço na memória principal? Então entra em ação o swapping;
- Retirar processos da memória para livrar espaço;
- Swap in;
- Swap out;

Swapping em memória virtual.