

[< VOLTAR](#)

Gerenciamento de Memória virtual

Com o avanço dos SO (Sistemas Operacionais), em alguns momentos a memória física torna-se insuficiente, precisando recorrer à memória virtual. Nessa aula iremos entender como isso é possível.

NESTE TÓPICO

- > Introdução
- > Referências

Marcar
tópico



Introdução

A memória principal sempre teve um papel importante nos processamentos, pois trata-se de uma importante área que armazena todos os processos (programas) que estão em processamento (em execução). Ela possui características específicas como, por exemplo a volatilidade, ou seja, ela libera automaticamente o espaço utilizado por um processamento assim que este termina.

O verbo utilizado para a “reserva” de espaço na memória é “Alocar”, exemplo: um espaço é alocado ou desalocado na memória.

Com objetivo de otimização dos recursos do computador, foi criada a Memória Virtual.

ATENÇÃO:

A Memória Virtual é uma extensão da memória principal, localizada no HD (Hard Disk).

Segundo Deitel (2005), sistemas de memória virtual dão aos processos a ilusão de que tem mais memória do que a contida no computador. Por esse motivo, há dois tipos de endereços em sistemas de memória virtual:

- Os referenciados por processos: chamados de endereços virtuais;
- Os disponíveis na memória principal: endereços físicos ou reais.

Para Machado (2007), o conceito de memória virtual fundamenta-se em não vincular o endereçamento feito pelo programa dos endereços físicos da memória principal. Dessa forma, programas e suas estruturas de dados deixam de estar limitados ao tamanho da memória física disponível, pois podem possuir endereços associados à memória secundária.

Espaço de Endereçamento Virtual

O conceito de memória virtual é muito parecido com o conceito de um vetor existente nas linguagens de programação. Quando um programa faz referência a um elemento do vetor, não há preocupação em saber a posição de memória, porque o compilador é responsável por gerar as instruções que implementam esse mecanismo.

Com relação aos endereços dos programas e dados, a memória virtual trabalha da mesma forma. A tarefa que estiver no ambiente de memória virtual não faz referência a endereços físicos de memória (endereços reais), mas apenas a endereços virtuais. Quando for chegado o momento da execução de uma instrução, o endereço virtual referenciado é traduzido para um endereço físico, pois o processador manipula apenas posições da memória principal (já tratamos disso anteriormente). O mecanismo de tradução do endereço virtual para endereço físico é denominado mapeamento.

Um processo é composto pelo contexto de hardware, contexto de software e espaço de endereçamento. Em ambientes que implementam memória virtual, o espaço de endereçamento do processo é conhecido como espaço de endereçamento virtual e representa o conjunto de endereços virtuais que o processo pode endereçar. (MACHADO, 2007).

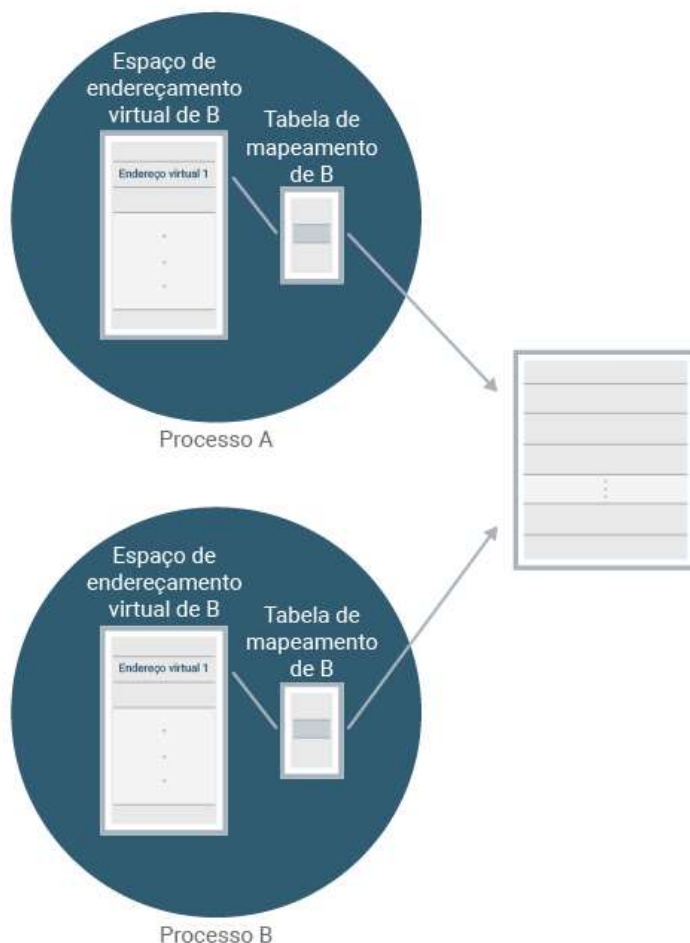
O programa, ao ser executado, apenas uma parte do seu código fica alocado na memória principal, permanecendo o restante na memória secundária até o momento de ser referenciado. Dessa forma, pode-se aumentar o compartilhamento da memória principal entre muitos processos, aumentando o grau da multiprogramação.

Para os desenvolvedores de software, não existe os endereços de memória virtual, os compiladores são responsáveis por gerar o código executável de acordo o espaço de endereçamento virtual, e o sistema operacional cuida dos detalhes durante sua execução.

Mapeamento

O processador só executa instruções que estejam alocados em endereçamentos reais, portanto, tem que existir uma rotina que transforme endereços virtuais em endereços reais. Esse é o conceito de mapeamento, que permite traduzir um endereço localizado no espaço virtual para um associado no espaço real. (MACHADO, 2007).

Para controlar o mapeamento, o sistema operacional necessita de uma tabela contendo a estrutura de dados de cada processo. Quando uma tarefa está em execução, o sistema utiliza a tabela de mapeamento dessa tarefa para fazer a tradução de seus endereços virtuais. Se por ventura outro processo entrar no estado de execução, o sistema operacional deverá referenciar a tabela de mapeamento do novo processo, isso só é possível por causa do uso de um registrador.



Mapeamento.

Fonte: MACHADO, F. B., MAIA, L. P.; Arquitetura de Sistemas Operacionais; 4a Ed., Rio de Janeiro: LTC; 2007.

Gerência de Memória Virtual

Para Machado (2007), memória virtual é uma técnica sofisticada e poderosa gerência de memória, em que as memórias principal e secundária são combinadas dando ao usuário a ilusão de existir uma memória muito maior que a capacidade real da memória principal.

O conceito de memória virtual fundamenta-se em não vincular o endereçamento feito pelo programa dos endereços físicos da memória principal. Dessa forma, programas e suas estruturas de dados deixam de estar limitados ao tamanho da memória física disponível, pois podem possuir endereços associados à memória secundária.

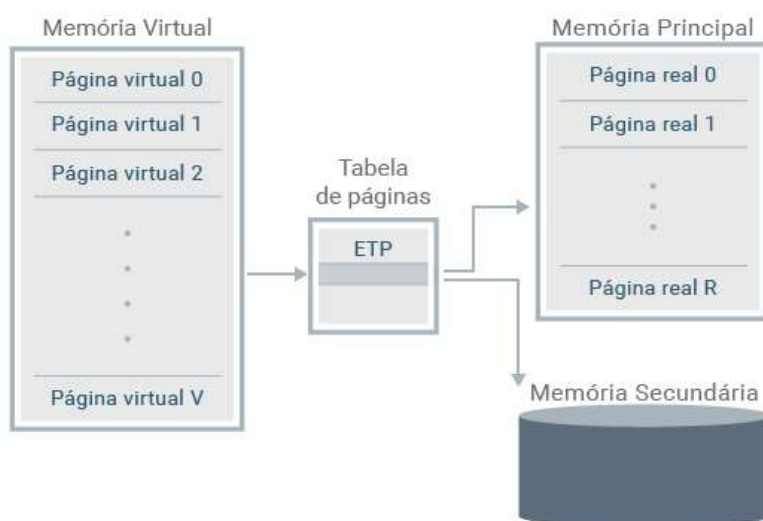
Memória Virtual por Paginação

Nesse algoritmo, os espaços de endereçamento virtual e o espaço de endereçamento real são divididos em blocos de mesmo tamanho denominados páginas.

Se estiverem no espaço virtual, as páginas são denominadas páginas virtuais, senão são chamadas de páginas reais ou frames.

Para transformar o endereço virtual em real, é necessário que os processos possuam as tabelas de páginas, e cada página virtual possua um apontamento de entrada na ETP (Entrada na Tabela de Páginas), com informações de mapeamento que irão possibilitar ao sistema localizar a página real correspondente.

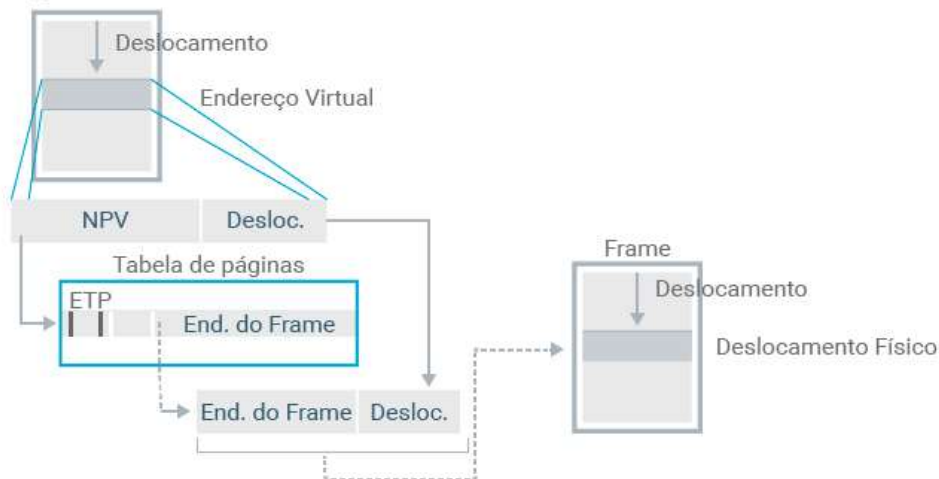
Quando um programa entra para ser executado, as páginas virtuais são transferidas da memória secundária para a memória principal (lembre-se de que o processo só executa instruções que estiverem alocadas na memória principal) e colocadas nos frames.



Paginação na Memória Virtual.

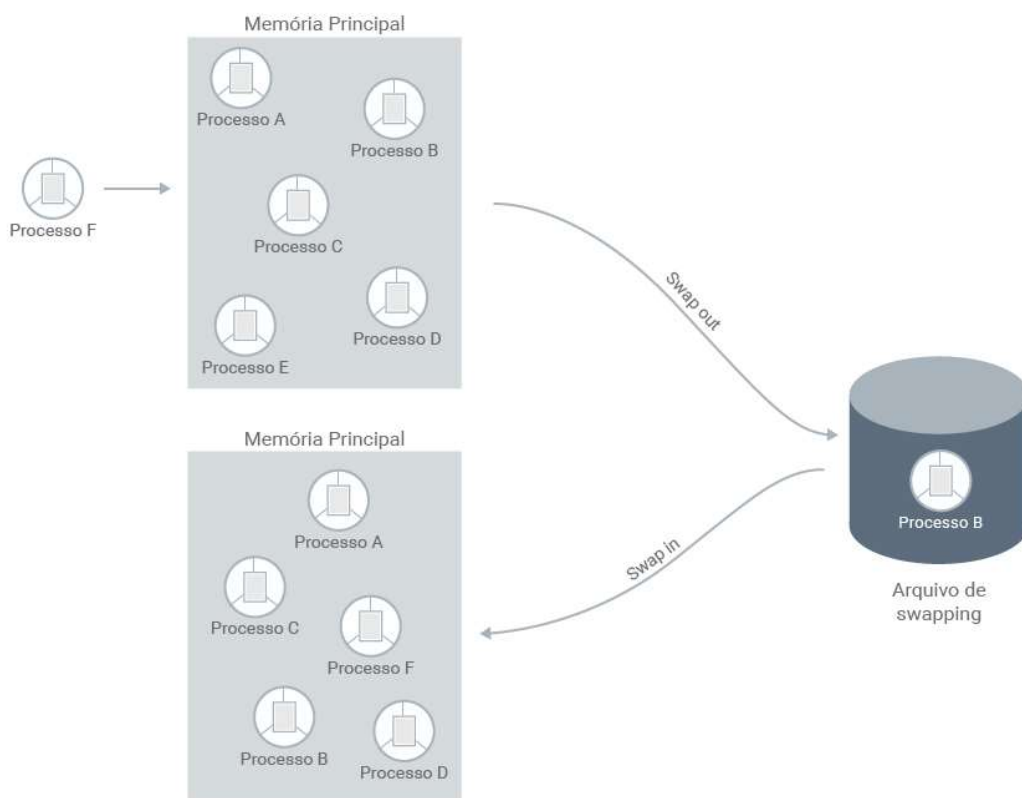
Fonte: MACHADO, F. B., MAIA, L. P.; Arquitetura de Sistemas Operacionais; 4a Ed., Rio de Janeiro: LTC; 2007.

Página virtual



Paginação na Memória Virtual.

Fonte: MACHADO, F. B., MAIA, L. P.; Arquitetura de Sistemas Operacionais; 4a Ed., Rio de Janeiro: LTC; 2007. Sistemas, Operacionais, SO, Memória, Virtual,



Swapping de Memória Virtual.

Fonte: MACHADO, F. B., MAIA, L. P.; Arquitetura de Sistemas Operacionais; 4a Ed., Rio de Janeiro: LTC; 2007.

Quando um programa faz solicitação a um endereço virtual, a rotina de mapeamento localizará na ETP (Entrada na Tabela de Páginas) da tabela do processo o endereço físico do frame no qual se encontra o endereço real correspondente. Dessa forma, o endereço virtual será formado pelo número

da página virtual (NPV) e por um deslocamento. O NPV identifica a página virtual que contém o endereço, agindo como um índice na tabela de páginas, o deslocamento, por sua vez, indica a posição do endereço virtual em relação ao início da página na qual se encontra. Logo o endereço físico é obtido combinando-se o endereço do frame, localizado na tabela de páginas, com o deslocamento, contido no endereço virtual.

A ETP possui outras informações, como por exemplo, o bit de validade (*valid bit*), que indica se uma página está ou não na memória principal. Se o bit tem o valor 0, está na página virtual, se é igual a 1, a página está localizada na memória principal.

Ao fazer referência a um endereço virtual, a unidade de gerência de memória checa, por meio do bit de validade, se a página que contém o endereço referenciado está ou não na memória principal (através do 0 e 1, citado no parágrafo anterior). Se a página não estiver na memória, é correto afirmar que ocorreu um *page fault*. Nessa situação, o sistema transfere a página da memória secundária (discos ou fitas) para a memória principal, realizando uma operação de E/S conhecida como *page in*, ou paginação.

A quantidade de *page faults* gerados por cada processo em um determinado intervalo de tempo é definida como taxa de paginação do processo.

Para Machado (2007), quando um processo referencia um endereço e ocorre um *page fault*, o processo em questão passa do estado de execução para o estado de espera, até que a página seja transferida do disco para a memória principal. Na troca de contexto, as informações sobre a tabela de mapeamento são salvas e as informações do novo processo escalonado são restauradas. Após a transferência da página para a memória principal, o processo é relocado na fila de processos no estado de pronto, e quando for reescalonado poderá continuar sua execução.

Swapping em Memória Virtual

A técnica de swapping, que segundo Machado (2007), é uma técnica que foi introduzida para contornar o problema de insuficiência de memória principal, aplicada à gerência de memória para programas que esperam por memória livre para serem executados, também pode ser aplicada em sistemas com memória virtual, permitindo aumentar o número de processos que compartilham a memória principal e, conseqüentemente, o grau de multiprogramação do sistema, que é a função dessa rotina.

Se existirem novos processos para serem executados e não houver memória principal livre suficiente para alocação, o sistema utiliza swapping, selecionando um ou mais processos para saírem da memória e oferecer espaço para novos processos. Depois de escolhidos, o sistema retira os processos da memória principal para a memória secundária (swap out), em que as páginas ou segmentos são gravados em um arquivo de swap (swap file). Com os processos salvos na memória secundária, os frames ou segmentos alocados são liberados para novos processos. Posteriormente, os processos que foram retirados da memória devem retornar para a memória principal (Swap in) para serem novamente executados.

Thrashing

Se o sistema operacional não possuir uma boa gerência de memória, haverá muitas ocorrências de programas mudando-se da memória principal para a secundária. Essa excessiva transferência de páginas / segmentos entre a memória principal e a memória secundária recebe o nome de thrashing. Encontraremos esse problema em sistemas que implementam tanto paginação como segmentação.

Na memória virtual por paginação, o thrashing ocorre em dois níveis:

- **Processo** – a excessiva de paginação ocorre em consequência do elevado número de page faults gerado pelo programa em execução. Esse problema faz com que o processo passe mais tempo esperando por páginas do que realmente sendo executado, o que determina o seu tempo de espera médio (TEM, visto na aula 11) maior do que o necessário, diminuindo a eficiência do sistema;
- **Sistema** – se houver concorrência entre os processos, ou seja, se existirem mais processos competindo por memória principal que no espaço disponível. A primeira solução é a redução do número de páginas de cada processo na memória: mas esse mecanismo leva ao thrashing de processo. Se redução não for suficiente, o sistema inicia o swapping, retirando os processos da memória principal para a memória secundária. Se esse mecanismo for levado ao extremo, o sistema passará mais tempo realizando swapping que atendendo outros processos, aumentando também o tempo de espera médio do sistema, comprometendo a produtividade e a eficiência do sistema operacional.

O thrashing em sistemas que implementam segmentação também ocorre em dois níveis:

- **Processo** – a transferência excessiva de segmentos é devida à modularidade extrema do programa;
- **Sistema** é semelhante ao da paginação, com a ocorrência do swapping de processos para liberar memória para os demais.

Este tópico finaliza aqui, esperamos que você tenha entendido como o SO lida com a memória virtual.

Bom estudo!

Quiz

Exercício

Gerenciamento de Memória virtual

INICIAR ➤

Quiz

Exercício Final

Gerenciamento de Memória virtual

INICIAR ➤

Referências

DEITEL, H. M.; DEITEL, P. J.; CHOFFNES, D. R. Sistemas operacionais. 3. ed. São Paulo: Pearson Prentice Hall, 2005.

MACHADO, F. B., MAIA, L. P.; Arquitetura de Sistemas Operacionais; 4a Ed., Rio de Janeiro: LTC; 2007.

TANENBAUM. Andrew. Sistemas operacionais modernos, New Jersey; Prentice Hall; 2a; 2008.



Avalie este tópico



ANTERIOR

Gerenciamento de Memória

Biblioteca

([https://www.uninove.br/conheca-](https://www.uninove.br/conheca-a-uninove/biblioteca/sobre-a-biblioteca/apresentacao/)

a-

uninove/biblioteca/sobre-

a-

biblioteca/apresentacao/)

Portal Uninove

(<http://www.uninove.br>)

Mapa do Site



Índice

Ajuda?
PRÓXIMO
(<https://ava.uninove.br/cursos/>)

Gerenciamento de Armazenamento

© Todos os direitos reservados

