

Resumo
- 50 -
2^o prova

MMU - Memory Management Unit

* Endereçamento Virtual:

tradução de endereço virtual p/ físico:

deslocamento
na página

- 1º - o MMU divide o endereço virtual em NPV e offset
Número de Página Virtual
- 2º - consultar a tabela de páginas usando o NPV p/ obter o nº da maldona física.
- 3º - combina o endereço da maldona física + o offset p/ gerar o endereço físico final.

► Técnicas de otimização → Tabelas multivivel e TLB:

↳ Tabelas multivivel: para evitar armazenar 2^{36} entradas (68,7bi) na RAM, usa-se hierarquia

- tabela de 1º nível (diretório): aponta p/ tabelas de 2º nível
- tabela de 2º nível: contém entradas de mapeamento página-maldona

↳ TLB (translation lookaside buffer): cache de traduções recentes p/ evitar acesso à tabelas de páginas em toda referência.

► Page Fault → se a página não estiver na RAM, o SO carrega do disco (área de swap).

► Proteção → bits de permissão (leitura, escrita, execução) em cada entrada da tabelas.

Alocação Contígua: é um modelo clássico de alocação de memória que atribui a um processo blocos de memória consecutivos (blocos de memória com endereços contínuos).

(geralmente)

↳ memória principal c/ duas partições:

- SO residente c/ o vetor de interrupções na parte baixa
- Os processos do usuário mantidos nos endereços altos da memória

↳ cada processo é contido em uma única região contígua de memória

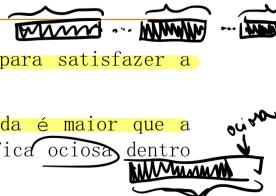
↳ registrador base tem o valor do menor endereço físico e o registrador limite tem o intervalo dos endereços lógicos (cada end. lógico deve ser menor que o registrador limite)

↳ MMU mapeia a endereço lógico dinamicamente

Quando um processo precisa ser executado, a memória é solicitada pelo processo. Se memória contígua suficiente for encontrada, o processo receberá memória alocada para iniciar sua execução. (caso contrário, é adicionado a uma fila de espera até ter memória).

Pode ocorrer fragmentação externa e baixa flexibilidade

Fragmentação



- ♦ Fragmentação externa - Existe memória para satisfazer a requisição, mas ela não é contígua.
- ♦ Fragmentação interna - A memória alocada é maior que a necessária; esta diferença de memória fica ociosa dentro da partição.
- ♦ Reduzir fragmentação externa por compactação
 - Deslocar o conteúdo da memória para que toda a memória livre fique contígua.
 - Compactação é possível apenas quando a relocação é dinâmica e feita em tempo de execução.

MMU - Unidade de Gerenciamento de Memória

↪ dispositivo hardware que mapeia o endereço virtual p/ o físico

O endereço virtual é o gerado pela CPU durante a execução de um programa. É o endereço que o programa "enxerga" e usa p/ acessar os dados.

O endereço físico é o local real e físico na memória RAM onde os dados estão realmente armazenados, e só pode ser acessado por meio da tradução do end. virtual para ele, feita pelo MMU.

O problema da fragmentação externa é quando, ao precisar armazenar um dado, não há espaço suficiente para essa requisição, mas ela não é contígua (contígua), estando separada em pequenos blocos espalhados pela memória.

Elas pode ser reduzida por meio de compactação \Rightarrow deslocar o conteúdo da memória p/ que toda memória livre fique contígua.

Quando os processos que precisam executar não cabem na memória ocorre o Swapping, em que o processo é removido temporariamente da memória p/ um disp. de memória auxiliar, e depois trazido de volta p/ retomar a execução.

Quando um processo excede o tamanho alocado de sua área de execução ocorre um estouro de buffer (overflow) gerando cenarios:

- compactação de memória adjacente
- etc

As formas delocar um bloco de memória p/ processos
não:

- First-Fit = o 1º bloco livre
- Best-Fit = o menor bloco livre
- Worst-Fit = o maior bloco livre

Questão 8. Considere um sistema cuja gerência de memória é feita através de partições variáveis. Inicialmente, existem os seguintes blocos: 10K, 4K, 20K, 18K, 7K, 9K, 12K e 13K, nessa ordem. Desenhe a memória e mostre como os blocos serão ocupados pelos processos de tamanho: 5K, 10K, 15K, 8K, 3K, 7K e 6K. Considere essa ordem de solicitação. Simule os seguintes algoritmos:

- a. First-fit
- b. Best-fit
- c. Worst-fit

a) First-fit

5K	1K	2K	3K	0	3K
10K	4K	20K	18K	7K	9K
5K	3K	10K	15K	7K	6K
8K					

b) Best-fit

0	3K	2K	1K	5K	7K
10K	4K	20K	18K	7K	9K
10K	15K	8K	7K	6K	
3K					

c) Worst-fit

3K	0	5K	8K	3K	5K
10K	4K	20K	18K	7K	9K
7K	5K	10K		3K	8K
15K				6K	

