

Gestão de Memória

Parte I - mecanismos

Sistemas Operativos

2011 / 2012

Sistemas Operativos – DEI - IST

Espaço de Endereçamento

- Conjunto de posições de memória que um processo pode referenciar
- E se referenciar outras posições de memória?
 - HW de gestão de memória desencadeia exceção
 - Tratada pelo SO (tipicamente termina processo)

Sistemas Operativos – DEI - IST

Hierarquia de Memória

- Memória principal (física ou primária):
 - acesso aleatório
 - tempo de acesso reduzido
 - custo elevado → reduzida dimensão
 - informação volátil
 - RAM + caches [+ registos]
- Memórias secundárias (ou de disco):
 - acesso aleatório (por blocos)
 - tempo de acesso elevado
 - custo reduzido → mais abundante
 - informação persistente

Gestão de Memória – Objectivo

- Gerir o espaço de endereçamento dos processos
 - assegurar que cada processo dispõe da memória que precisa
 - garantir que cada processo só acede à memória a que tem direito (protecção)
 - otimizar desempenho dos acessos



Breve Incursão pelo Endereçamento Real

- Motivos para a incursão:
 - Terminologia
 - Históricos (e.g., MS-DOS)
 - Sistemas embebidos (e.g., smart-cards) usam endereçamento real

Sistemas Operativos – DEI - IST



Endereços Reais vs. Virtuais

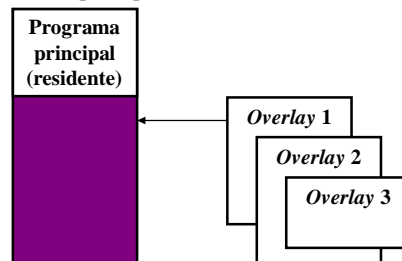
- Endereçamento Real
 - Endereço indicado no programa é aquele que é acedido na memória principal
 - Usados pelos sistemas iniciais monoprogramados
 - Limitações?
 - Dimensão dos programas?
 - Multiprogramação?
 - Capacidade de ser executado em diferentes máquinas?

Sistemas Operativos – DEI - IST

Mecanismo de Sobreposição (Overlay)

- Possibilidade de executar programas com dimensão superior à memória principal em sistemas com endereçamento real
- Os overlays continuam a ter de ser dimensionados para a memória física disponível.
- Programador indica explicitamente quando deve ser carregado um overlay

Memória principal

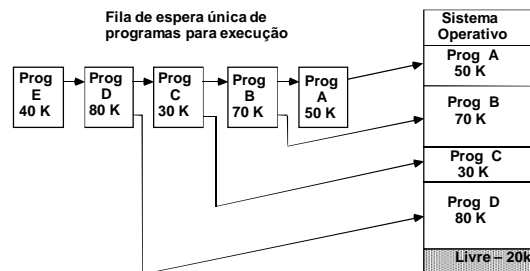


Sistemas Multiprogramados com Endereçamento Real – Partições Fixas

- Memória dividida em partições de dimensão fixa
- Em cada partição é carregado um programa
- Programas recolocáveis
 - Endereçamento baseado
 - Registo base + deslocamento
- Vários programas em execução implica mecanismos de protecção
 - Registo limite → validado em cada acesso
- Fragmentação interna: dimensão dos programas não coincide exactamente com a dimensão das partições

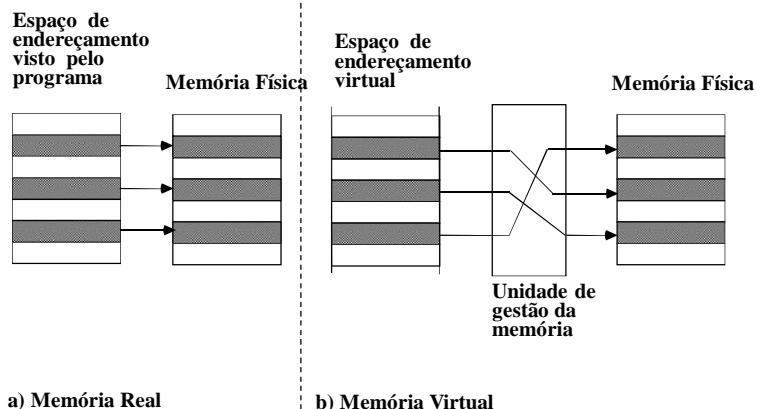
Sistemas Multiprogramados com Endereçamento Real – Partições Variáveis

- Fragmentação externa: muitas partições de dimensão muito reduzida
- Quando um programa termina, a sua partição pode ser associada a outra (se for contígua), criando uma única de maior dimensão
- Implica a recompactação da memória → processamento suspenso



Sistemas Operativos – DEI - IST

Endereços Reais vs. Virtuais



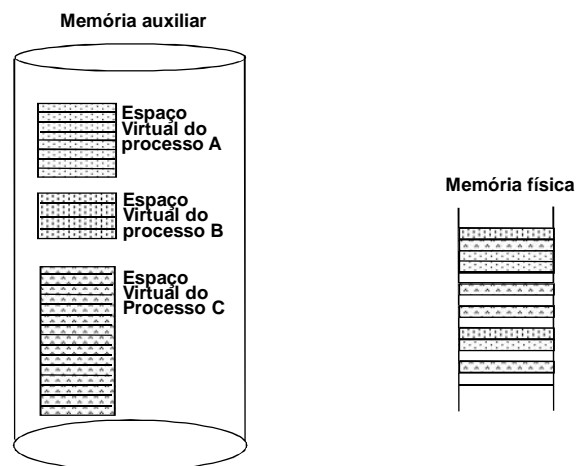
Sistemas Operativos – DEI - IST

Endereçamento Virtual

- Espaço de endereçamento dos processos não linearmente relacionado com a memória física
- Endereços virtuais são sempre convertidos (pela UGM) para endereços reais
- Para minimizar a informação necessária à conversão, a memória virtual é logicamente dividida em blocos contíguos:
 - Endereço virtual = (bloco, deslocamento)
- Dois tipos de blocos:
 - Segmentos - dimensão variável.
 - Páginas - dimensão constante.
- Alguns blocos podem não residir em memória principal

Sistemas Operativos – DEI - IST

Espaço de Endereçamento Virtual



Sistemas Operativos – DEI - IST



Princípio da Localidade de Referência

- Em que consiste?
- Porque motivo torna a gestão em blocos eficiente?

Sistemas Operativos – DEI - IST

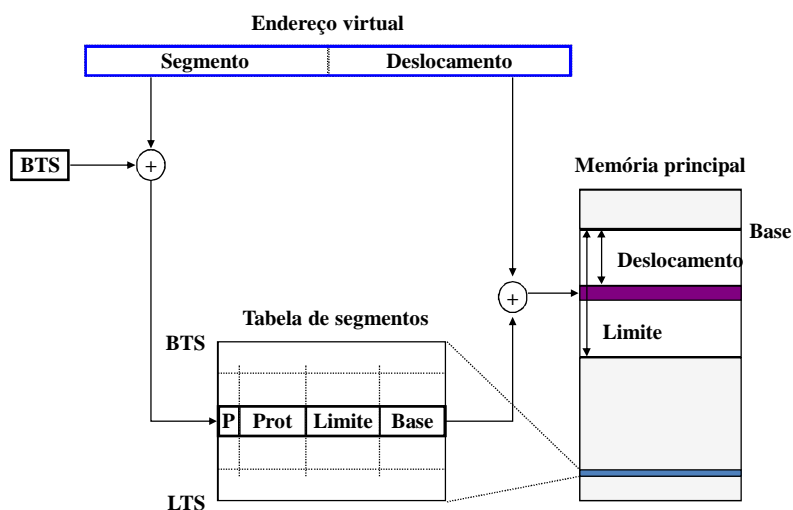


Segmentação

- Divisão dos programas em segmentos lógicos que reflectem a sua estrutura funcional:
 - rotinas, módulos, código, dados, pilha, etc.
 - a conversão de endereços virtuais é linear em cada segmento
 - o programador pode ter que se preocupar com a gestão de memória quando escreve um programa
- Gestão de memória suporta abstracções das linguagens de programação → Segmento é a unidade de:
 - carregamento em memória (eficiência)
 - protecção
- Dimensão dos segmentos é limitada pela arquitectura e não pode exceder a dimensão da memória principal

Sistemas Operativos – DEI - IST

Tradução de Endereços Virtuais em Memória Segmentada



Sistemas Operativos – DEI - IST

Memória Virtual Segmentada

- Fragmentação: externa ou interna?
- Protecção:
 - verificação de limites de endereçamento intra-segmentos
 - verificação e limitação dos tipos de acesso ao segmento: leitura, escrita e execução
 - processos diferentes têm tabelas de segmentos diferentes: espaços de endereçamento disjuntos e inacessíveis a terceiros
- Partilha de memória entre processos:
 - basta colocar nas tabelas de segmentos dos processos em questão o endereço real do segmento a partilhar
 - os endereços virtuais usados para aceder ao segmento partilhado podem ser diferentes nos vários processos
 - a protecção dum segmento partilhado é definida para cada processo através da respectiva tabela de segmentos

Sistemas Operativos – DEI - IST

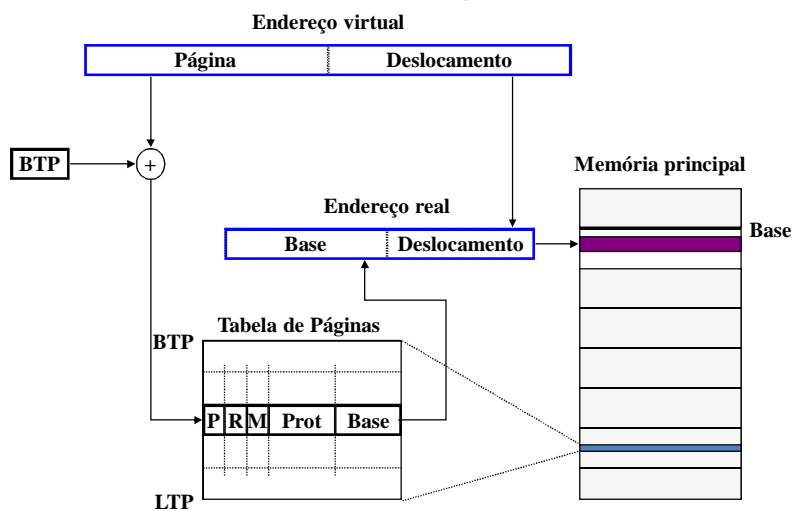
Paginação

- Espaço de endereçamento virtual de dimensão superior à da memória principal:
 - o programador não “vê” a gestão de memória
- Na memória principal mantêm-se algumas páginas
 - Restantes carregadas de memória secundária quando necessário (falta de página)
- As instruções do processador têm de ser recomeçáveis
- A dimensão das páginas (constante) é normalmente muito menor que a da memória principal e influencia:
 - A fragmentação (externa ou interna?)
 - O número de faltas de páginas
 - Tempo da sua resolução (transferência)
 - A dimensão das tabelas de páginas e listas de páginas mantidas pelo sistema operativo

Memória Virtual Paginada (cont.)

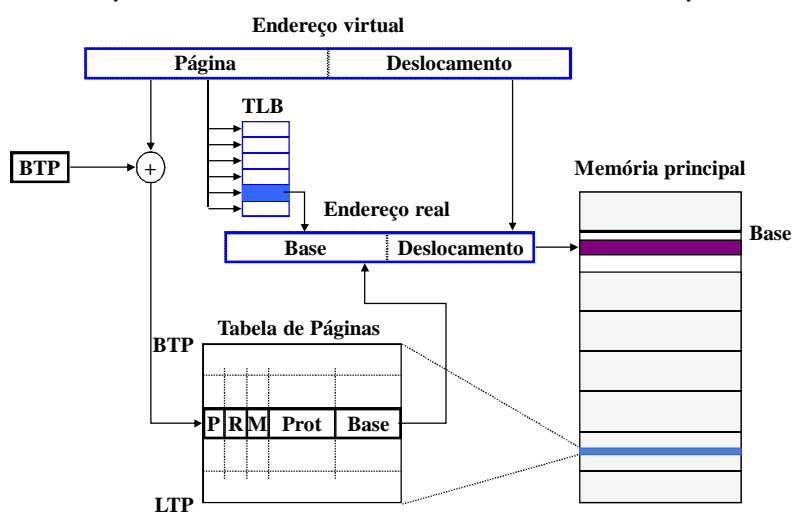
- Protecção:
 - Verificação dos tipos de acesso: leitura, escrita e execução.
 - Processos diferentes têm tabelas de páginas diferentes: espaços de endereçamento disjuntos e inacessíveis a terceiros
- Partilha de memória entre processos:
 - Semelhante ao usado para memória segmentada
 - Partilha de blocos lógicos: partilha de múltiplas páginas
 - Não é possível partilhar menos que uma página (versus arquitectura segmentada → partilha de uma divisão lógica do programa)

Tradução de Endereços Virtuais em Memória Paginada

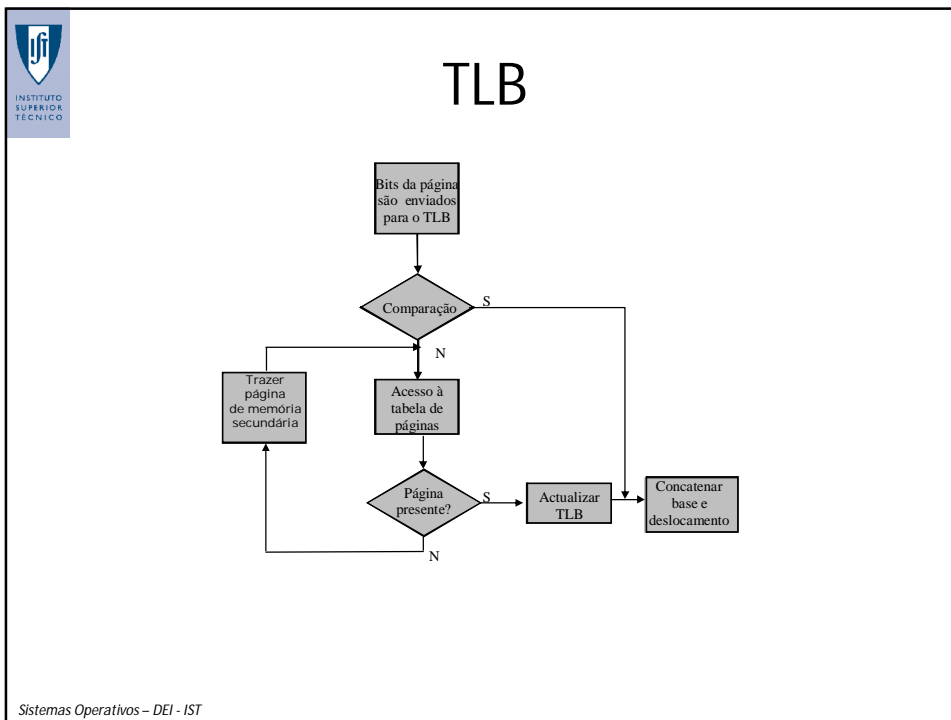


Sistemas Operativos – DEI - IST

Tabela de Tradução de Endereços (Translation Lookaside Buffer, TLB)



Sistemas Operativos – DEI - IST



TLB

- O ideal seria guardar na TLB não as últimas, mas as próximas páginas a que o programa irá aceder.
- Como isso é impossível de prever:
 - toma-se o funcionamento recente do programa como uma boa previsão para o que ele fará no futuro próximo.
 - se um programa acedeu a uma página, é expectável que os próximos acessos sejam dentro da mesma página.
- A dimensão destas tabelas é pequena, em geral (64, 128 entradas), pois o seu custo é elevado:
 - a sua dimensão é cuidadosamente testada, usando simulações e dados estatísticos de programas existentes, de forma a obter percentagens de sucesso muito elevadas (90-95%)
 - um factor que também influencia a dimensão da TLB é o *quantum* dos processos.
 - sendo a TLB limpa em cada comutação de processos, quanto maior for o *quantum*, maior é o número de páginas acedidas, o que leva à necessidade de ter mais entradas na TLB.

Sistemas Operativos – DEI - IST



Memória Virtual Paginada – Problema

- Qual a dimensão da tabela de páginas com endereços virtuais de 32 bits e páginas de 4 kBytes?
- (Ainda pior se fossem endereços de 64 bits...)

Sistemas Operativos – DEI - IST



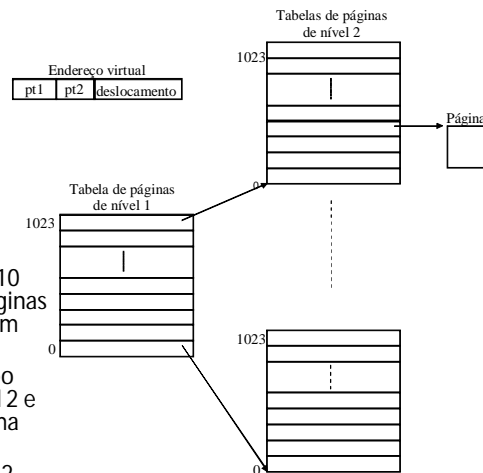
Memória Virtual Paginada – Problema

- endereços de 32 bits, páginas de 4 Kbytes:
 - espaço de endereçamento de cada processo engloba mais de um milhão de páginas ($2^{32}/2^{12}=2^{20}$ páginas)
 - no máximo, a tabela de páginas correspondente ocupa 4 Mbytes (assumindo 4 bytes para cada PTE e 2^{20} PTE, teremos $2^2 \cdot 2^{20}=2^{22}$).
- organizar tabela de páginas de duas formas para minimizar o espaço ocupado:
 - tabela de páginas multinível
 - tabela de páginas invertida

Sistemas Operativos – DEI - IST

Tabelas de Páginas Multi-Nível

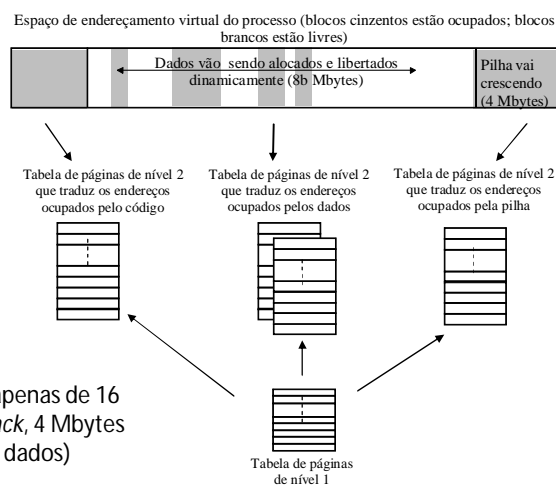
- várias tabelas de páginas (em vez de apenas uma) que se referem entre si de forma hierárquica.
- endereço virtual é dividido em vários grupos de bits:
 - correspondem a sucessivas entradas nas tabelas de páginas da hierarquia e, finalmente,
 - ao deslocamento dentro de uma dada página.
- Assumindo 12 bits para o deslocamento, 10 bits para pt1 e 10 bits para pt2, temos páginas com 4 Kbytes e cada tabela de páginas com 1024 entradas.
- Cada entrada na tabela de páginas de topo referencia uma tabela de páginas de nível 2 e cada entrada nestas tabelas referencia uma página.
- Portanto, cada tabela de páginas de nível 2 referencia 1024 páginas, ou seja, 4 Mbytes.



Sistemas Operativos – DEI - IST

Tabelas de Páginas Multi-Nível

- Vantagem:
 - apenas estão em memória as tabelas de páginas correspondentes às páginas que estão de facto a ser utilizadas pelo processo correspondente.
- Ex.:
 - processo que necessita apenas de 16 Mbytes (4 Mbytes de stack, 4 Mbytes de código e 8 Mbytes de dados)



Sistemas Operativos – DEI - IST

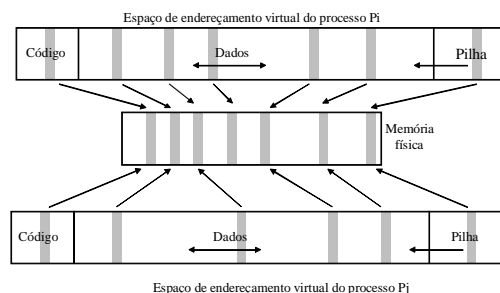
Tabela de Páginas Invertida

- Mecanismo de tradução de endereços virtuais em endereços reais que vimos até agora baseia-se em:
 - entrada numa tabela de páginas para cada página virtual.
 - para sistemas com endereços de 32 bits isso resulta, no máximo (i.e. quando todo o espaço está ocupado) em mais de um milhão de entradas.
 - actualmente, já há processadores que suportam endereços com 64 bits pelo que, com páginas de 4 Kbytes, teremos uma tabela de páginas com 2^{52} entradas ($2^{64}/2^{12}=2^{52}$).
 - uma vez que cada entrada na tabela de páginas tem, no mínimo, 8 bytes (64 bits), a dimensão da tabela de páginas correspondente terá mais de 60 milhões de Gbytes ($2^{52} \times 2^4 = 2^{56} = 2^{30} \times 2^{20} \times 2^6 = 1 \text{ Gigabyte} \times 1 \text{ Megabyte} \times 64$)
- Num computador com endereços virtuais de 64 bits, páginas de 4 Kbytes, e 1 Gbyte de memória primária teríamos, no máximo:
 - 262144 páginas físicas (i.e. $2^{30}/2^{12}=2^{18}$) e 2^{52} páginas virtuais.
- Enorme diferença entre o número de páginas virtuais e o número de páginas físicas sugere que, nestes sistemas:
 - a tradução de endereços virtuais em endereços reais deve ser feita usando tabelas em que cada entrada referencia uma página física e não uma página virtual.
 - este é o conceito subjacente à tabela de páginas invertida.

Sistemas Operativos – DEI - IST

Tabela de Páginas Invertida

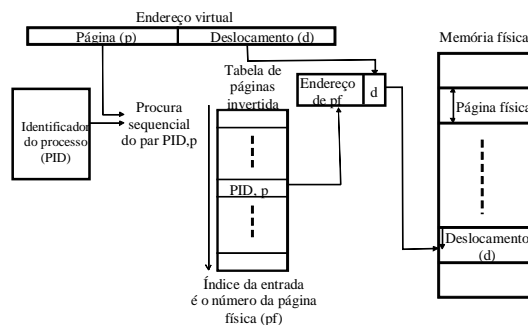
- Cada entrada numa tabela de páginas invertida contém, para a página física correspondente:
 - a identificação da página virtual respectiva e do processo de cujo espaço de endereçamento essa página faz parte.
 - esta tabela é única para o sistema, i.e. não existe uma para cada processo como nas soluções anteriormente apresentadas.
- Note-se que, ao longo do tempo, para cada página física podem existir várias páginas virtuais que nela são mapeadas



Sistemas Operativos – DEI - IST

Tabela de Páginas Invertida

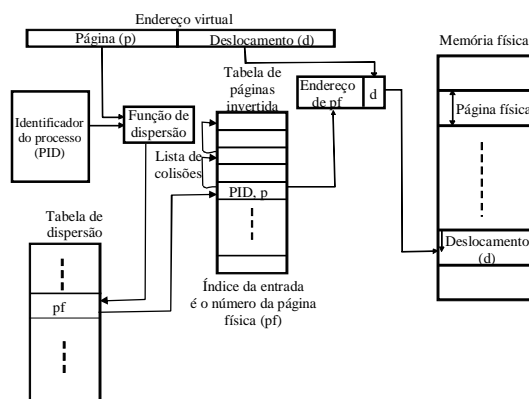
- Na tabela de páginas invertida, a entrada i corresponde à página física i
- Assim, o endereço físico de cada página é imediatamente conhecido.
- Devido ao facto de para cada página física, em diferentes momentos, poderem existir várias páginas virtuais que nela são mapeadas:
 - cada entrada guarda a identificação do processo a cujo espaço de endereçamento pertence a página virtual correspondente



Sistemas Operativos – DEI - IST

Tabela de Páginas Invertida

- Estando a tabela ordenada por páginas físicas, é possível que toda a tabela tenha de ser percorrida à procura do par (PID, p) em causa.
- Esta tradução tem de ser efectuada sempre, e não apenas quando há uma falta de página:
 - extremamente importante que seja efectuado de forma muito rápida de modo a não comprometer o desempenho do sistema.
- Pode ainda ser complementado com uma TLB:
 - assumindo que esta contém a informação de tradução relativa às páginas mais utilizadas, a obtenção do endereço real correspondente a um endereço virtual efectua-se com a rapidez necessária para assegurar o bom desempenho do sistema



Sistemas Operativos – DEI - IST

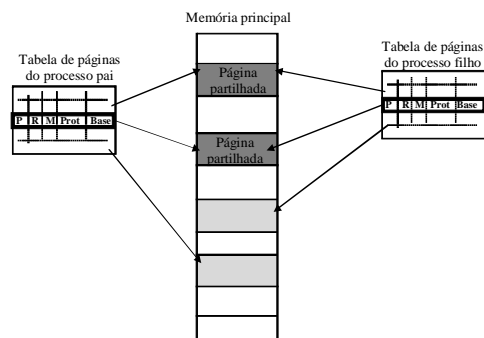
Memória Partilhada

- Para partilhar uma zona de memória entre vários processos basta ter:
 - nas tabelas de páginas dos processos em causa, um conjunto de PTE indicando, cada uma, o mesmo endereço físico.
- Como para a protecção, este mecanismo é mais complicado e menos elegante na memória paginada que na segmentada.
- Caso interessante de partilha de memória:
 - quando um processo-pai cria um processo-filho (por exemplo, através da chamada fork em Unix).
 - o espaço de endereçamento do processo-filho é criado à imagem e semelhança do processo-pai.
 - de modo a tornar a criação de processos o mais eficiente possível, o sistema operativo suporta um mecanismo denominado *copy-on-write* (COW).

Sistemas Operativos – DEI - IST

Memória Partilhada - COW

- o sistema operativo tem de distinguir:
 - acesso em escrita a uma das páginas do tipo *copy-on-write*
 - acesso em escrita a uma página protegida que não seja desse tipo (i.e. que resulta de um acesso efectivamente não autorizado)
- uma excepção a uma página *copy-on-write* diferencia-se pelo facto de:
 - a página em causa estar protegida para escrita mas
 - o grupo de páginas do qual faz parte permitir esse mesmo tipo de acesso.



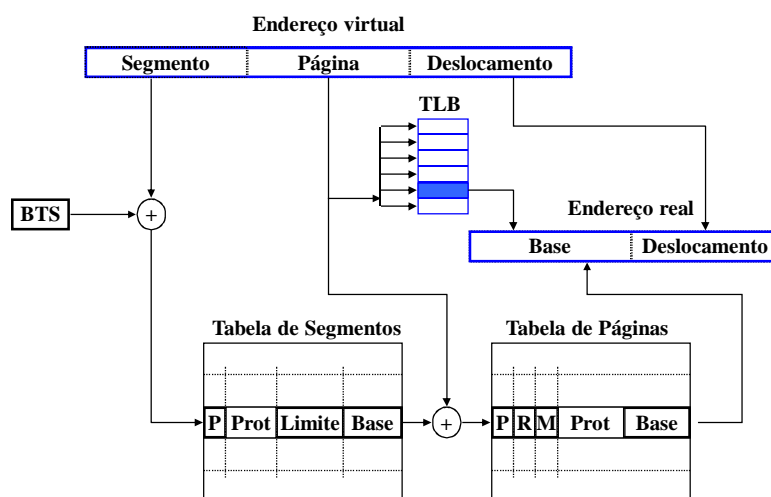
Sistemas Operativos – DEI - IST

Memória Partilhada - COW

- Páginas partilhadas entre processos distintos levanta um potencial problema de desempenho:
 - quando é alterado o estado de uma página partilhada, por exemplo, deixa de estar carregada em memória passando para memória secundária,
 - o núcleo do sistema operativo deve actualizar todas as PTE correspondentes em todos os processos de cujo espaço de endereçamento essa página faz parte.
- Um modo de efectuar esta actualização é o seguinte:
 - percorrer todas as tabelas de páginas de todos os processos à procura das PTE que referenciam a página partilhada e actualizá-las em conformidade.
 - esta é uma solução muito ineficiente
- Os sistemas operativos actuais têm outras soluções que asseguram melhor desempenho.:
 - quando abordarmos o Linux e o Windows, as soluções passam por usar mecanismos denominados **reverse mapping e prototype PTE**, respectivamente
- Consistem em:
 - usar estruturas de dados auxiliares associadas a páginas partilhadas
 - para qualquer uma dessas páginas físicas partilhadas, as estruturas de dados auxiliares permitem saber quais as PTE que a referenciam, sem ter de percorrer as tabelas de páginas de todos os processos.

Sistemas Operativos – DEI - IST

Memória Segmentada/Paginada



Sistemas Operativos – DEI - IST

Unix e Linux

Unix

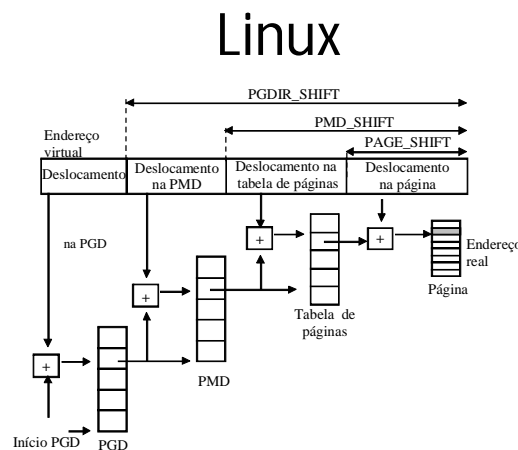
- As primeiras implementações do Unix (versão 7 e anteriores) executavam-se no PDP-11:
 - tinha uma arquitectura segmentada com 16 bits
 - o espaço de endereçamento dos processos era de 64 Kbytes, dividido em oito segmentos de 8 Kbytes cada.
- As versões com gestão de memória segmentada estiveram essencialmente relacionadas com implementações simplificadas para microprocessadores, em particular o sistema Xenix para o 80286
- A gestão de memória era muito simples:
 - os programas eram carregados na sua totalidade em memória
 - caso não houvesse espaço disponível em memória, o sistema operativo transferia para memória secundária os processos que estivessem bloqueados ou com menor prioridade
 - A transferência de processos era feita por um processo denominado *swapper*.

Unix

- As versões actuais do Unix (e similares):
 - realizam-se principalmente sobre arquitecturas paginadas.
- A versão Unix 3BSD foi a primeira a suportar memória virtual paginada tal como a conhecemos hoje
- Na evolução introduzida pelo Unix V:
 - o espaço de endereçamento era dividido em regiões.
- Um processo tinha três regiões:
 - código, dados e pilha.
- Novas regiões podiam ser criadas dinamicamente durante a execução dos programas.
- Cada região continha uma tabela de páginas própria e uma tabela de descritores de blocos de disco com a informação de localização das páginas quando estas estavam em memória secundária.

Sistemas Operativos – DEI - IST

- O primeiro conjunto (deslocamento na PGD) identifica uma entrada na PGD do processo em causa; esta entrada na PGD contém o endereço físico inicial de uma PMD;
- Na PMD é identificada uma entrada em particular através do deslocamento dado pelo segundo conjunto de bits (deslocamento na PMD);
- O conteúdo da entrada na PMD contém o endereço físico de uma tabela de páginas; nesta é identificada uma entrada em particular através do deslocamento dado pelo terceiro conjunto de bits (deslocamento na tabela de páginas);
- Por último, esta entrada contém o endereço inicial da página física que inclui o byte cujo endereço virtual está a ser traduzido; para obter o endereço físico do byte em causa, basta adicionar o deslocamento indicado pelo quarto conjunto de bits (deslocamento na página).



- tabelas de páginas multinível com três níveis
- tabela de mais alto nível é designada por Page Global Directory (PGD)
- as tabelas de nível intermédio são designadas por Page Middle Directory (PMD)

Sistemas Operativos – DEI - IST