Cap. 4 - Gerenciamento de Memória no UNIX

- **objetivo:** com base nos conceitos de gerenciamento de memória anteriormente apresentados, detalhamos como se dá o gerenciamento de memória no UNIX.

- Versões anteriores ao *System V* e ao 4.2 BSD empregavam um esquema de permuta *(swapping)* de processos como política de gerenciamento de memória;

- ... versões atuais empregam o esquema de paginação por demanda.

4.1 - Paginação por Demanda

- tabela de páginas;
- chamada *fork* num sistema paginado;
- chamada *exec* num sistema paginado.

4.2 - Processo Paginador

4.3 - Falta de Página

4.4 - Falta de Proteção

4.1 - Paginação por Demanda

- Este esquema necessita da habilidade do *hardware* de reiniciar uma instrução interrompida pela ausência de página cujo endereço foi referenciado na instrução;

- ... assim que a página é trazida para a memória, a intrução interrompida é reiniciada.

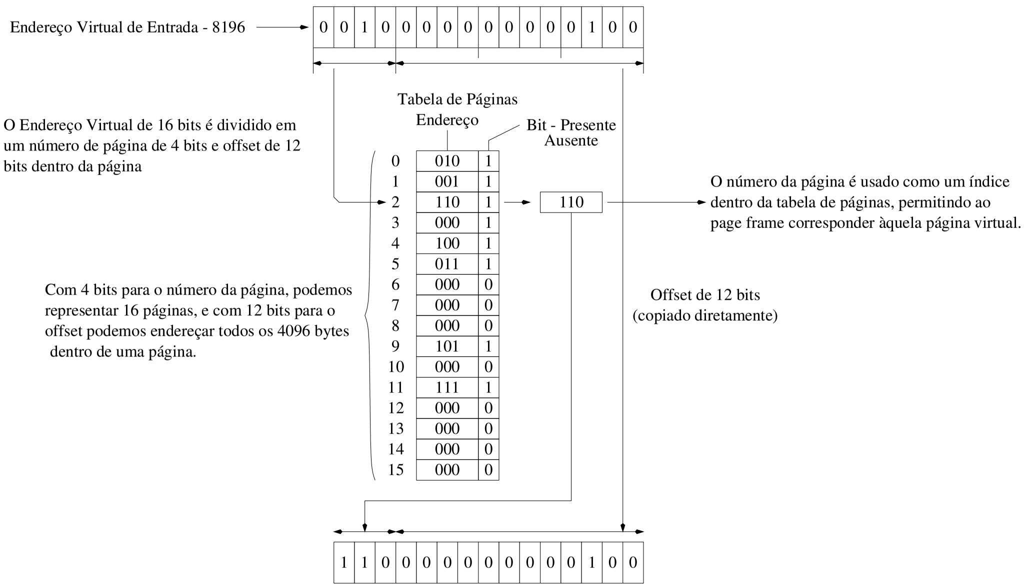
- ... no esquema de paginação por demanda, o espaço virtual de endereçamento é muito superior à quantidade de memória física disponível, sendo limitado apenas pelo capacidade de endereçamento virtual da MMU;

- ... o núcleo mantém 4 estruturas principais para fins de gerenciamento de memória:

❶ tabela de páginas
❸ descritores de blocos
❷ tabela de frames (pfdata)
❹ tabela de uso de *swap*

4.1.1 - Tabela de Páginas

- ... tabela tem dimenão fixa pois a quantidade de páginas é igual à quantidade física de memória dividida pelo tamanho da página.



... 4.1.1 - Tabela de Páginas

★ ... cada entrada na tabela de páginas possui os seguintes campos:

- **endereço físico** de memória que contém os dados referentes à esta página;
- **idade da página**, ou seja, por quantos ciclos esta página está ativa (na memória);
- flag COPY-ON-WRITE indicando se esta página está sendo compartilhada para fins de leitura, com a ressalva de que deve ser desmembrada caso algum processo que a compartilha altere seu conteúdo;
- **flag de modificação** que indica alteração recente do conteúdo da página;
- **flag de referência** indicando alguma referência recente ao conteúdo da página;
- **flag de validade** indicando se o conteúdo da página é válido, ou seja, o endereço físico guarda o conteúdo da página que foi referenciada;
- **flag de proteção** indicando se o conteúdo da página referenciada permite operações só de leitura (READ ONLY) ou leitura/escrita (READ/WRITE);

4.1.2 - Descritores de Bloco

✯ ... **descritor de bloco** com os seguintes campos:

- dispositivo de *swap* com área disponível para ler/gravar o conteúdo da página;
- número do bloco alocado à página;
- o tipo da página: *swap*, arquivo executável,

✍ ... este campo também está presente na tabela de páginas.

4.1.3 - Tabela de Frames

✯ ... a tabela de *frames* armazena dados adicionais à página:

- endereço físico de memória que contém os dados referentes à esta página;
- contador de referência indicando quantos processos compartilham esta página;
- dispositivo de *swap* associado à página;
- número do bloco alocado para a página.

4.1.4 - Tabela de Swap

- ... finalmente a tabela de *swap* é acessada pelo dispositivo de *swap* bem como pelo número do bloco neste dispositivo;

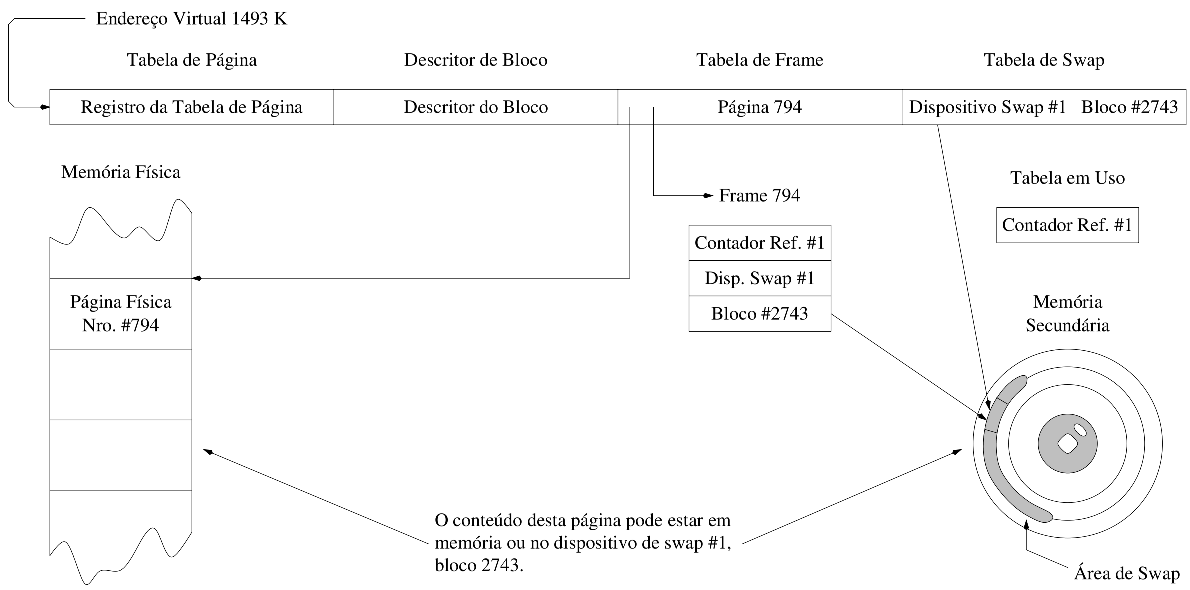
- ... esta tabela armazena apenas um contador de referência indicando quantas páginas se utilizam deste bloco em disco referente a esse dispositivo.

✍ deve-se notar que algumas informações são replicadas em tabelas distintas;

- ... esta replicação tem por objetivo beneficiar a eficiência do esquema de paginação, diminuindo o número de consultas às tabelas.

4.1.5 - Exemplo de Referência a um Endereço

- como forma de ilustrar as várias estruturas de dados empregadas no gerenciamento de memória, considere uma referência ao endereço virtual 1493K;



4.1.6 - Chamada “fork” num Sistema Paginado

- ... após uma chamada de sistema **fork** num sistema paginado, o cenário estabelecido apresenta algumas particularidades, são elas:

❶ a área de texto não é duplicada tendo tanto o processo pai quanto o filho as mesmas entradas para a tabela de páginas (estas páginas são do tipo READ ONLY);

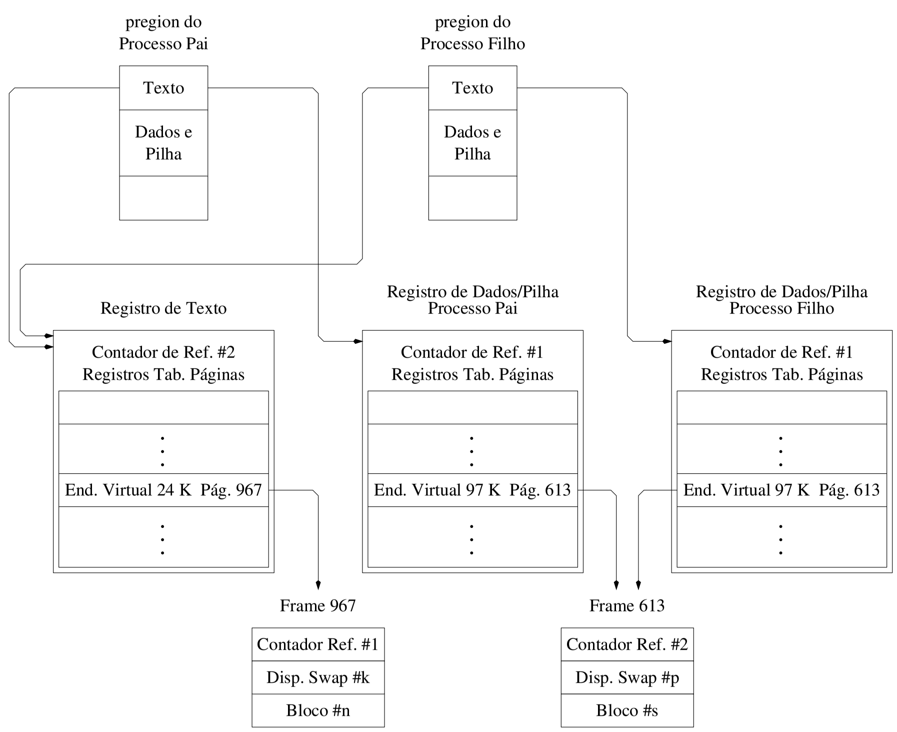
❷ o núcleo duplica as tabelas de páginas que contém as áreas de dados e de pilha;

❸ as entradas destas tabalas compartilham as mesmas entradas na tabela de frames (com o contador de referência agora em 2);

❹ todas as entradas na tabela de páginas são marcadas como COPY-ON-WRITE;

- ... a razão para isto é que quando um dos dois processos alterar o conteúdo da página, a mesma deve ser desmembrada, desvinculando-se os endereços físicos de memória e as entradas na tabela de frames.

... 4.1.6 - Chamada “fork” num Sistema Paginado



4.1.7 - Chamada "exec" num Sistema Paginado

✯ ... durante uma chamada **exec**, o núcleo carrega o programa executável do disco para a memória e executa os seguintes passos:

❶ ... num sistema paginado, pode ocorrer que o tamanho do executável supere o tamanho físico da memória e, neste caso, após todo o executável ter sido carregado, parte dele já se encontra no dispositivo de *swap*;

❷ ... monta-se a tabela de páginas com os respectivos descritores de blocos para o processo, dado que o sistema conhece o tamanho do executável a priori, informação esta presente no cabeçalho do próprio executável;

❸ ... as páginas são marcadas como *demand zero* para áreas de pilha ou *demand fill* para àreas de texto e dados e na sequência o núcleo começa a cópia das regiões de texto e de dados para as páginas atribuídas em memória física;

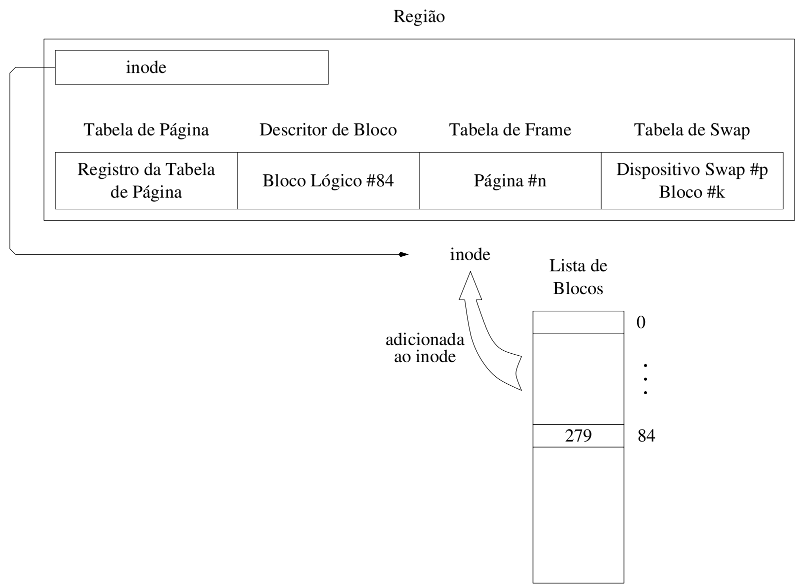
- ... o núcleo aloca uma página para cada registro da tabela de página;

... 4.1.7 - Chamada "exec" num Sistema Paginado

- ... páginas *demand zero* (para pilha) não estão presentes no código executável, sendo simplesmente alocadas e zeradas e páginas *demand fill* (texto e dados) são copiadas do disco das respectivas porções do executável;
- ... para copiar diretamente do executável para uma página em memória, o núcleo adiciona ao *inode* (em memória) um vetor de blocos que compõem o executável;
- ... o descritor de bloco da tabela de página, nesta fase, armazena o índice do bloco no vetor que contém a porção do executável a ser carregado nesta página;
- ... ao copiar a porção do executável para a página, o núcleo localiza o bloco acessando seu índice no descritor de bloco e seu número na posição correspondente no vetor que contém a porção do executável.

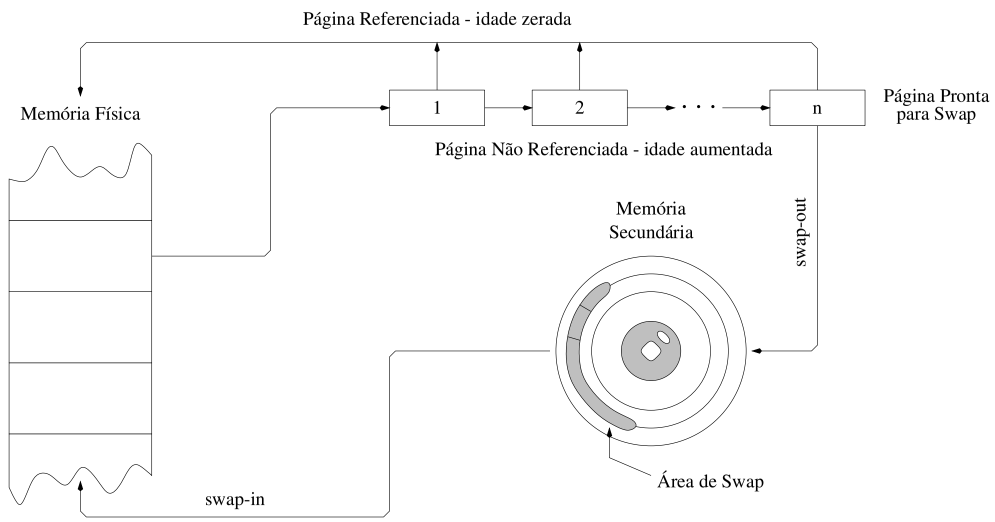
... 4.1.7 - Chamada “exec” num Sistema Paginado

- ... uma vez copiado a parte do arquivo executável para a memória física, o descritor de bloco é atualizado, contendo agora um bloco físico de *swap* associado à página.



4.2 - Processo Paginador

- **função** - remover da memória páginas não referenciadas por um longo tempo;
- ... quando executado, o processo monta uma lista de páginas candidatas à permuta e, adicionalmente, quando da troca o bit de referência é zerado;



... 4.2 - Processo Paginador

- ... uma página é candidata à permuta quando seu contador de referência foi zerado há um determinado número de passadas do processo paginador;
- ... se uma página candidata à permuta é novamente referenciada, a mesma é removida da lista de páginas candidatas à permuta;
- ... com esse intuito, o processo paginador é ativado quando a quantidade de memória disponível atinge um limite mínimo e neste caso páginas são removidas da memória e gravadas em disco até que um limiar de memória livre seja atingido;
- ... o gravar uma página em disco, o processo paginador apaga o bit de validade da página e decrementa seu contador de referência na tabela de frames;

... 4.2 - Processo Paginador

- ... se o contador de referência vai a zero (indicando que um único processo estava utilizando a página), o núcleo adiciona o campo da tabela de frames referente à página numa lista de páginas livres;
- ... o conteúdo de uma página na lista de páginas livres continua válido até que o sistema associe a página à outro processo;
- ... mesmo na lista de páginas livres, uma página pode ser revalidada (sem necessidade de trazê-la do disco) caso um processo torne a referenciá-la. Neste caso, a página é removida da lista de páginas livres, tendo seu bit de validade reativado.

4.3 - Falta de Página

- **falta de página** - estado gerado por um processo quando a referência é feita a uma página cujo bit de validade esteja zerado;

- ... neste caso, o núcleo deve providenciar a validação da página para que o processo possa continuar sua execução e, nesse sentido, uma página referenciada pode se encontrar em um dos 05 estados abaixo descritos:

❶ no dispositivo de *swap* e não em memória;

❷ em memória principal, na lista de páginas livres;

❸ num arquivo executável sendo carregado;

❹ marcada como *demand fill*;

❺ marcada como *demand zero*.

... 4.3 - Falta de Página

❶ ... no primeiro caso, o sistema deve alocar uma página livre e atualizar o conteúdo da memória com aquele presente no dispositivo de *swap* e, assim, o processo é bloqueado até que a operação de entrada/saída seja completada.

❷ ... no segundo caso, caso a página desde quando adicionada à lista livre não foi associada a nenhum outro processo, a página continua válida, sendo removida da lista de páginas livres sem que nenhuma operação de E/S se faça necessária.

❸ ... no terceiro caso, o núcleo atráves do descritor de bloco encontra no vetor de blocos do *inode* aquele que contém a parte do executável sendo requisitado;

- ... uma operação de entrada/saída se faz necessária para trazer a página do disco para a memória, causando também uma associação de página a um bloco de *swap*;

❹ ... uma página é alocada, sendo seu conteúdo atualizado como no caso anterior.

❺ ... uma página é alocada e seu conteúdo de memória é simplesmente zerado.

4.4 - Falta de Proteção

- **falta de proteção** - ocorre quando um processo acessa uma página válida, mas os bits de proteção da página impedem o acesso;

- ... 02 exemplos são típicos nesta situação:

❶ ... o processo tenta escrever numa página que permite somente leitura, ou seja, marcada como READ ONLY (p. ex., área de texto);

❷ ... o processo tenta escrever numa página marcada como COPY-ON-WRITE, por exemplo, após uma chamada **fork**.

... 4.4 - Falta de Proteção

❶ no primeiro caso, a falta de proteção é ilegal, sendo um sinal (tipicamente SIGBUS) enviado ao processo;

- ... o processo é então terminado, caso não tenha provido gerenciador para o sinal;

❷ no segundo caso, a falta de proteção é legal, mas o processo é suspenso até o núcleo desmembrar a página que foi referenciada;

- ... O núcleo providencia a alocação de uma nova página, marca-a como READ/WRITE, copia o conteúdo da página em falta para a nova página, e decrementa seu contador de referência;

- ... se o contador vai a zero, a página marcada como COPY-ON-WRITE pode ser reusada. A tabela de página do processo aponta para a nova página, agora com permissao de escrita. A partir deste ponto, o processo retoma sua execução.