

# Sistemas Operacionais

**Prof. Dr. Helder Oliveira**

# Plano de Aula

- Memória virtual
- Paginação

# Memória virtual

- Registradores de base e registradores limite.
  - Podem ser usados para criar abstração de espaços de endereçamentos.
  - Como gerenciar o bloatware?
    - Softwares que usam muita memória.
- O problema dos programas maiores do que a memória existe desde o início da computação.
  - 1960 – dividir em módulos pequenos (**sobreposição**)

# Sobreposição

- Carregava apenas o gerenciador de sobreposições.
- Imediatamente carregava e executava a sobreposição 0.
- Quando terminava, ele dizia ao gerenciador de sobreposições para carregar a sobreposição 1, acima da sobreposição 0 na memória (se houvesse espaço para isso), ou em cima da sobreposição 0 (se não houvesse).
- Sistemas de sobreposições eram altamente complexos.
  - Muitas sobreposições na memória ao mesmo tempo.
- Sobreposições - Mantidas no disco e transferidas para dentro ou para fora da memória pelo gerenciador de sobreposições.

# Sobreposição

- Sistema operacional
  - Realizava a troca de sobreposições do disco para a memória e vice-versa.
- Programador
  - Divisão do programa em módulos manualmente.
  - Tarefa cansativa, chata e propensa a erros.
  - Poucos programadores eram bons nisso.
- Alguém resolveu passar todo o trabalho para o computador.
- **Memória virtual**

# Memória virtual

- Cada programa tem seu próprio espaço de endereçamento, o qual é dividido em blocos chamados de páginas.
- Cada página é uma série contígua de endereços.
  - Páginas são mapeadas na memória física.
  - Nem todas precisam estar na memória física ao mesmo tempo para executar o programa.

# Técnicas de Memória Virtual

- **Paginação**
  - Blocos de tamanho fixo
  - O espaço de endereçamento virtual é dividido em páginas virtuais.
- **Segmentação**
  - Blocos de tamanho arbitrário chamados de segmentos.
  - Contém mesmo tipo de informações.
- Mapeamento entre endereços reais e virtuais (MMU)
- Muitos SOs usam uma mistura das duas técnicas.

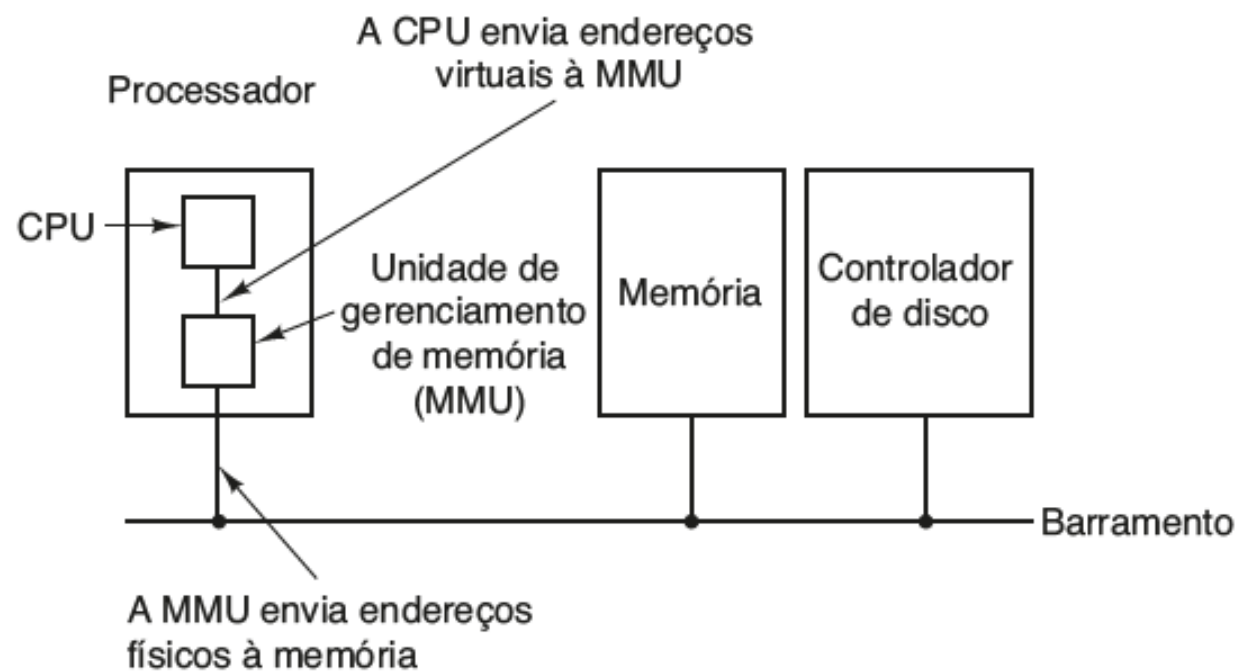
# Paginação

- Técnica utilizada pela maioria dos sistemas de **memória virtual**.
- Programas referenciam um conjunto de endereços de memória.
- Endereços podem ser gerados usando indexação, registradores base, registradores de segmento e outras maneiras.
- Endereços gerados por computadores são **endereços virtuais**.
- Endereços virtuais formam o **espaço de endereçamento virtual**.
- Computadores sem memória utiliza o barramento de memória.
- Usando a Memória virtual - os endereços de memória vão para a unidade de gerenciamento de memória (MMU) .

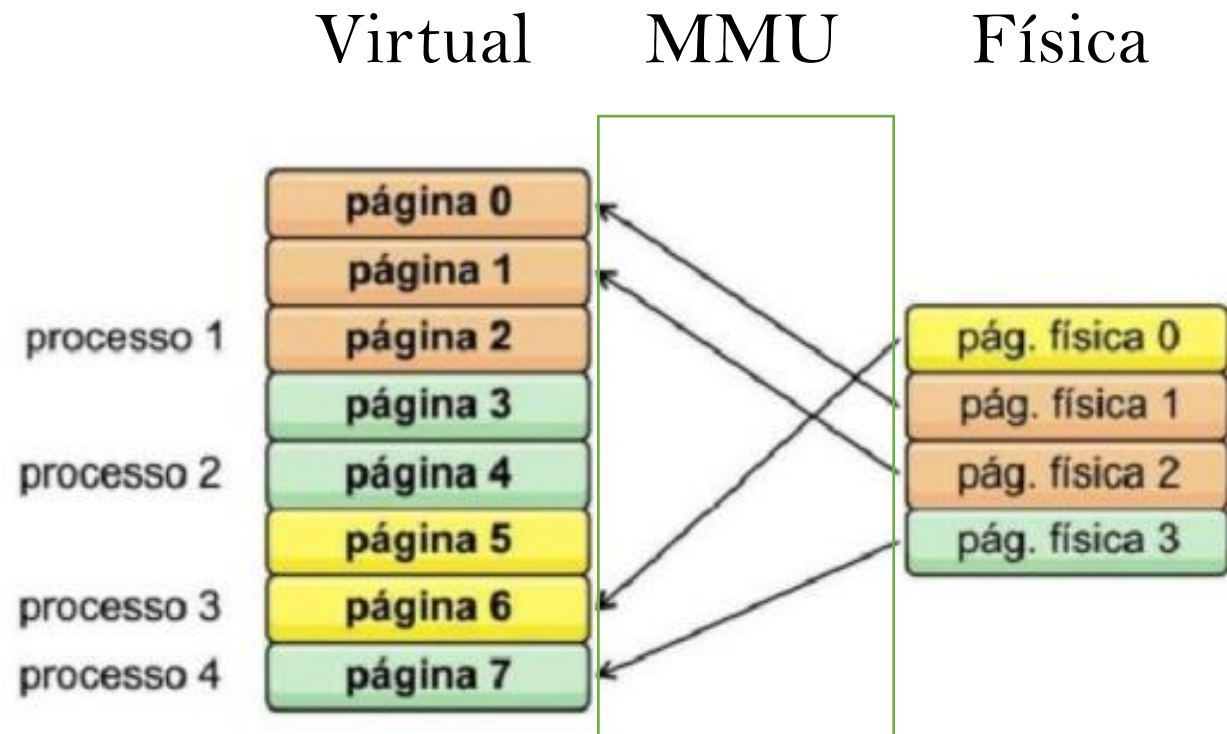


# Unidade de gerenciamento de memória -MMU

**FIGURA 3.8** A posição e função da MMU. Aqui a MMU é mostrada como parte do chip da CPU porque isso é comum hoje. No entanto, logicamente, poderia ser um chip separado, como era no passado.

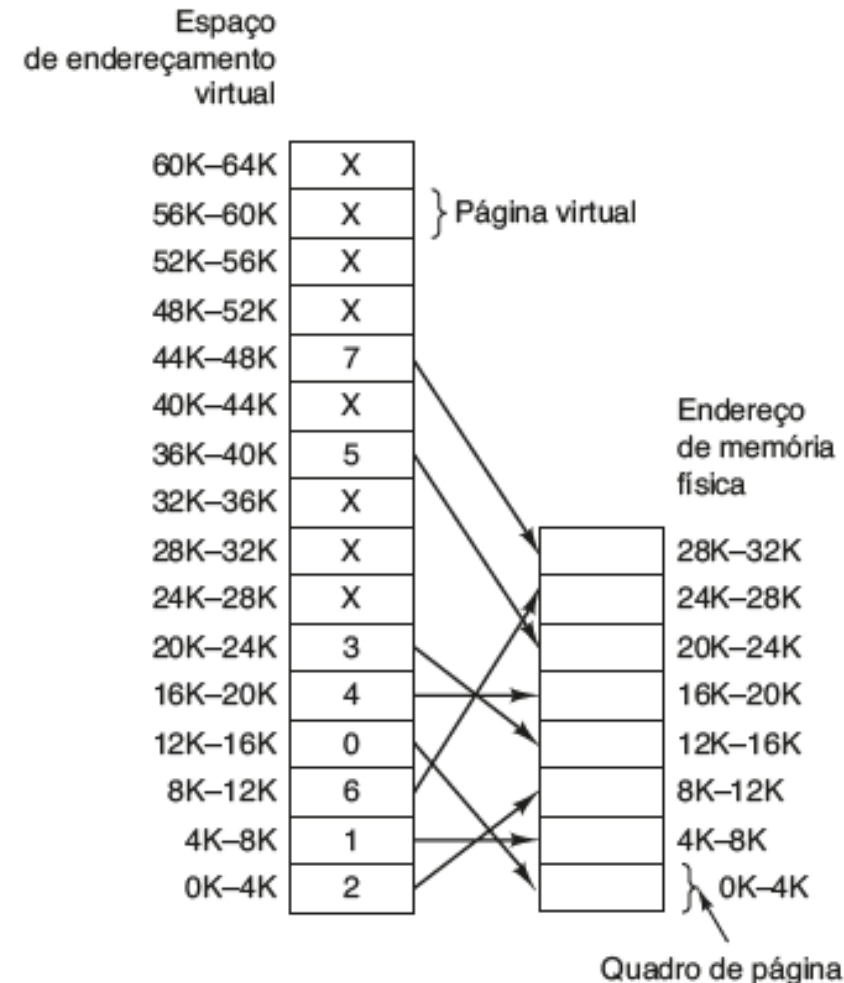


# Memória Virtual



# Endereços virtuais

**FIGURA 3.9** A relação entre endereços virtuais e endereços de memória física é dada pela **tabela de páginas**. Cada página começa com um múltiplo de 4096 e termina 4095 endereços acima; assim, 4K a 8K na verdade significa 4096-8191 e 8K a 12K significa 8192-12287.

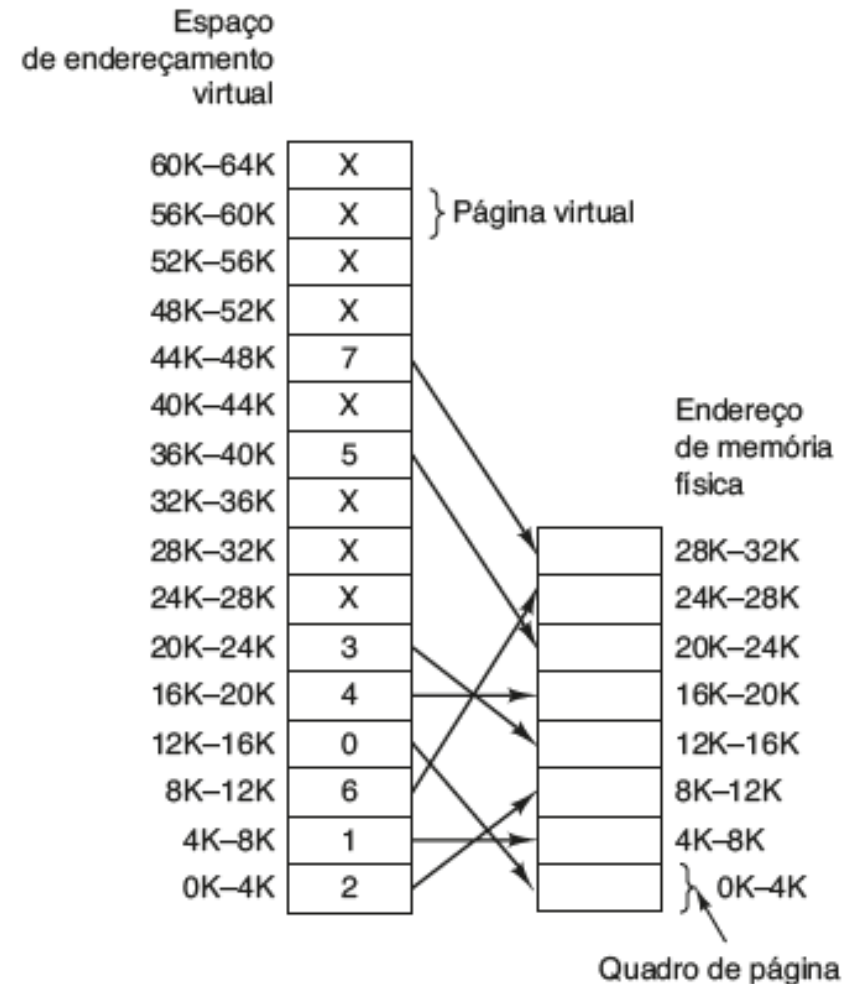


# Espaço de endereçamento

- O espaço de endereçamento virtual consiste em unidades de tamanho fixo chamadas de **páginas**.
- As unidades correspondentes na memória física são chamadas de **quadros de página (frames)**.
- Transferências entre a memória RAM e o disco são sempre em páginas inteiras.
- Muitos processadores dão suporte a múltiplos tamanhos de páginas que podem ser combinados e casados como o sistema operacional preferir.

# Endereços virtuais

**FIGURA 3.9** A relação entre endereços virtuais e endereços de memória física é dada pela **tabela de páginas**. Cada página começa com um múltiplo de 4096 e termina 4095 endereços acima; assim, 4K a 8K na verdade significa 4096-8191 e 8K a 12K significa 8192-12287.

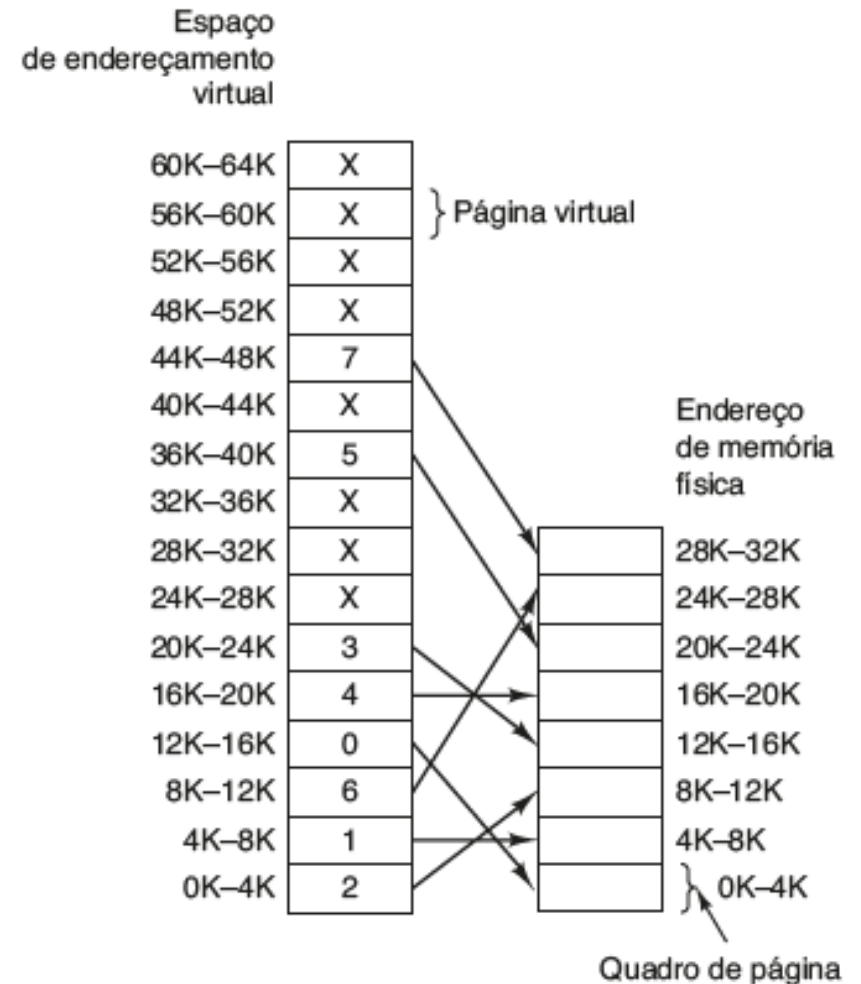


# Paginação

- A habilidade de mapear 16 paginas virtuais em 8 quadros não soluciona o problema de espaço de endereçamento virtual.
- Exemplo anterior:
  - Oito quadros de páginas físicas, mapeiam apenas oito das páginas virtuais na memória física.
  - As outras paginas virtuais, mostradas com um X na figura, não estão mapeadas.
  - No hardware real, um bit Presente/ausente controla quais páginas estão fisicamente presentes na memória.

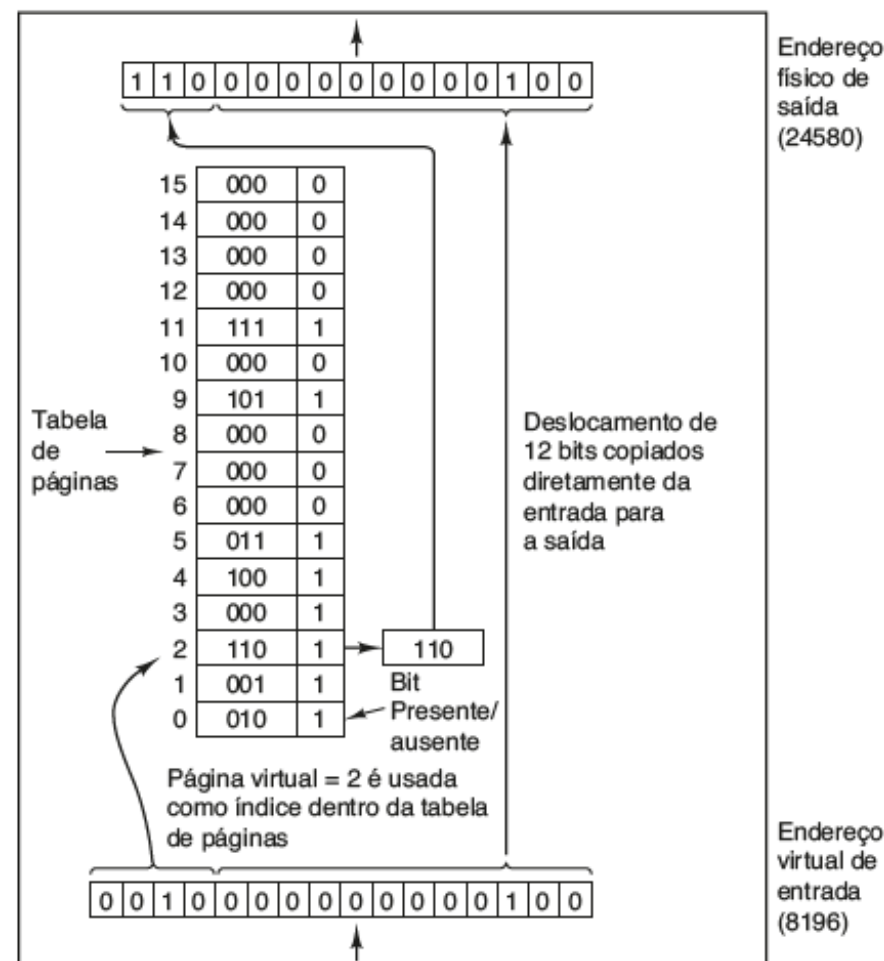
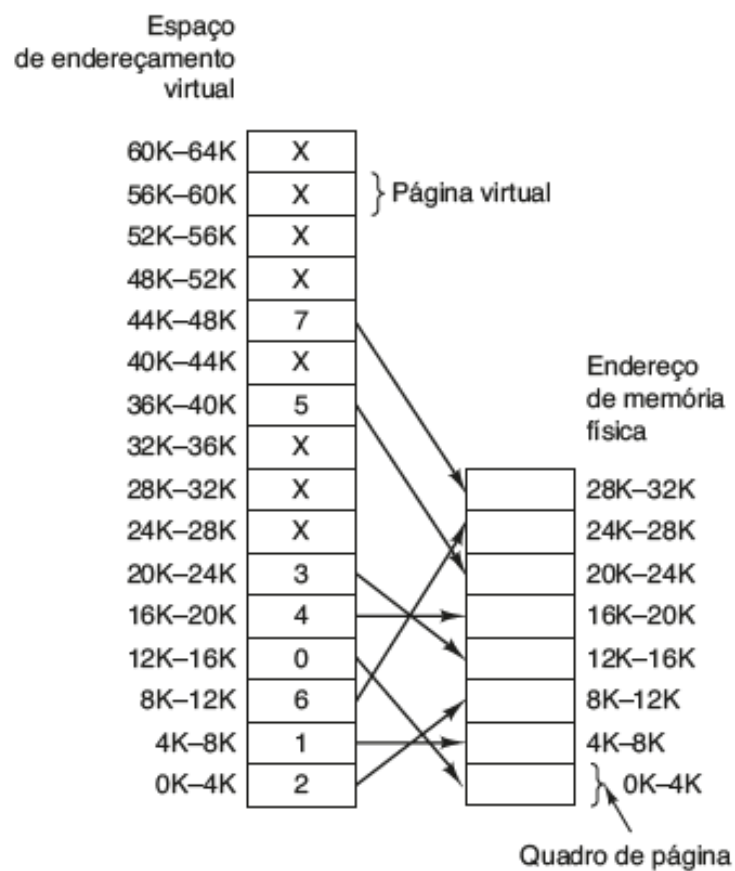
# Endereço não mapeado – falta de página

- O que acontece se o programa referencia um endereço não mapeado?
- MMU observa que a página não está mapeada e faz a CPU desviar para o sistema operacional.
- **Falta de página.**
- O sistema operacional escolhe um quadro de página pouco usado e escreve seu conteúdo de volta para o disco (se já não estiver ali).



# Dentro da MMU

**FIGURA 3.10** A operação interna da MMU com 16 páginas de 4 KB.





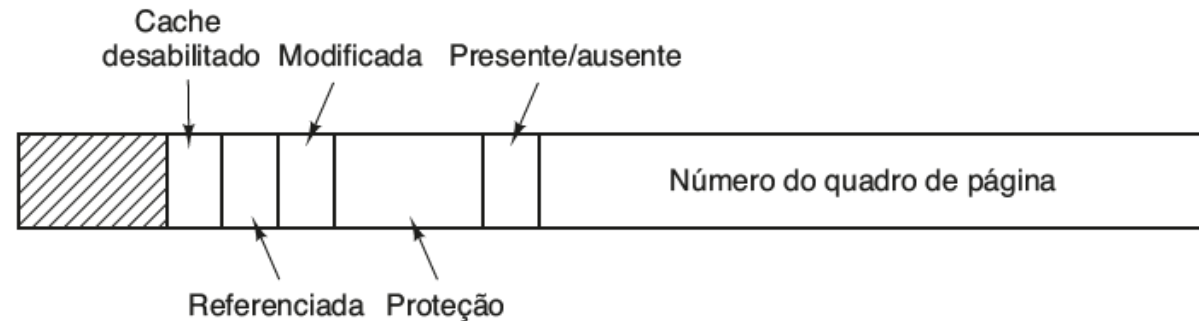
# Tabelas de páginas

- O objetivo da tabela de páginas é mapear as páginas virtuais em quadros de páginas.
- Matematicamente falando, é uma função com o número da página virtual como argumento e o número do quadro físico como resultado. Usando o resultado dessa função, o campo da página virtual em um endereço virtual pode ser substituído por um campo de quadro de página, desse modo formando um endereço de memória física.

# Tabelas de páginas

- Estrutura de uma entrada da tabela de páginas:

**FIGURA 3.11** Uma entrada típica de uma tabela de páginas.



- O tamanho varia de computador para computador, mas 32 bits é um tamanho comum.
- O campo mais importante é o Número do quadro de página. Afinal, a meta do mapeamento de páginas é localizar esse valor.

# Acelerando a paginação

- Em qualquer sistema de paginação, duas questões fundamentais precisam ser abordadas:
  - O mapeamento do endereço virtual para o endereço físico precisa ser rápido.
  - Se o espaço do endereço virtual for grande, a tabela de páginas será grande.
- O projeto mais simples (pelo menos conceitualmente) é ter uma única tabela de página consistindo de uma série de registradores de hardware rápidos, com uma entrada para cada página virtual, indexada pelo número da página virtual.

# TLB ou memória Associativa

- Pequeno dispositivo de hardware para mapear endereços virtuais para endereços físicos sem ter de passar pela tabela de páginas.

**FIGURA 3.12** Uma TLB para acelerar a paginação.

Válida	Página virtual	Modificada	Proteção	Quadro de página
1	140	1	RW	31
1	20	0	R X	38
1	130	1	RW	29
1	129	1	RW	62
1	19	0	R X	50
1	21	0	R X	45
1	860	1	RW	14
1	861	1	RW	75

# Gerenciamento da TLB por software

- **Ausência leve** (soft miss): ocorre quando a página referenciada não se encontra na TLB, mas está na memória.
- **Ausência completa** (hard miss): ocorre quando a página em si não está na memória (e, é claro, também não está na TLB).
  - **Passeio na tabela de páginas** (page table walk): procurar o mapeamento na hierarquia da tabela de páginas.

# Tabelas de páginas para memórias grandes

- **Tabelas de páginas multinível:** evitar manter todas as tabelas de páginas na memória o tempo inteiro. Em particular, aquelas que não são necessárias não devem ser mantidas.
- **Tabelas de páginas invertidas:** alternativa para os níveis cada vez maiores em uma hierarquia de paginação. Nesse projeto, há apenas uma entrada por quadro de página na memória real, em vez de uma entrada por página de espaço de endereço virtual. São comuns em máquinas de 64 bits porque mesmo com um tamanho de página muito grande, o número de entradas de tabela de páginas é gigantesco.

# Leitura

- SISTEMAS OPERACIONAIS MODERNO 4ª edição
  - 3.3 Gerenciamento de Memória

Dúvidas?