

Segmentação de Memória

Pedro Cruz

EEL770 – Sistemas Operacionais

- Algoritmos de substituição
 - Substituição de páginas não usadas recentemente
 - Substituição de páginas primeiro a entrar, primeiro a sair
 - Substituição de páginas segunda chance
 - Substituição de páginas do relógio
 - Substituição de páginas usadas menos recentemente
 - Substituição de páginas não usadas frequentemente
 - Substituição de páginas do conjunto de trabalho
 - Substituição de páginas *WSClock*
- Modos de substituição
 - Global
 - Local

Questões de implementação



- Tarefas de paginação
- Tratamento de falta de página
- Recuperação de instrução
- Retenção de páginas
- Armazenamento de apoio
- Separação de política e mecanismo

Tarefas de paginação



- Criação do processo
 - Tamanho do processo (em número de páginas)
 - Tabela de páginas
- Execução do processo
 - *Memory Management Unit* (MMU)
 - *Translation Lookaside Buffer* (TLB)
- Falta de página
 - Recuperação de página
- Término do processo
 - Liberar páginas
 - Liberar tabelas

Criação do processo



- Decidir tamanho do processo
 - Em páginas
- Criar a tabela de páginas
 - Em memória, se processo estiver sendo executado
- Alocar espaço em disco
 - Quadros de página do processo
 - Código
 - Pode ser o próprio arquivo executável
 - Dados
- Adicionar informações de memória à tabela de processos

Tratamento de falta de página



- Interrupção gerada por hardware
 - Rotina salva registradores
 - Rotina chama sistema operacional
- Página faltante descoberta por SO (endereço virtual)
 - Registradores
 - Instrução
- Verificação de proteção e existência da página
- Substituição de página
 - Limpeza de quadro de página
- Escalonamento de processos
- Recuperação do processo original

Recuperação de instrução



- Instruções estão em mais de um endereço
 - Contador de programa nem sempre estará no início da instrução quando ocorrer a falta de página
- Exemplo
 - Se falha de página ocorrer nos operandos, PC não necessariamente será 1000
- Solução
 - CPUs possuem um “PC” que aponta o início da instrução

Endereço	Valor	Legenda
1000	MOVE	Código da instrução
1002	6	Primeiro operando
1004	1	Segundo operando

Retenção de páginas



- Algumas páginas devem ser retidas
 - Compartilhadas entre processos
 - Usadas para fazer E/S
 - Hardware especial cuida de E/S (DMA)
 - Atua em paralelo à CPU
 - Recebe um endereço pra ler/escrever E/S
 - Ignora paginação

Armazenamento de apoio



- Páginas ficam armazenadas em disco
 - Espaço em disco deve ser alocado
 - Se área especial em disco
 - Mapeamento automático
 - Perda de espaço em disco
 - Crescimento de processos complicado
 - » Áreas diferentes para pilha, código e dados
 - Se áreas alocadas dinamicamente
 - Deve haver uma tabela de mapeamento
 - Ganho de espaço em disco
- Páginas podem estar em arquivos
 - Código pode ser o próprio arquivo executável

Armazenamento de apoio

Quadros em área especial de disco
1
2
3
4
5
6
7

Quadros dinâmicos
2
4
5

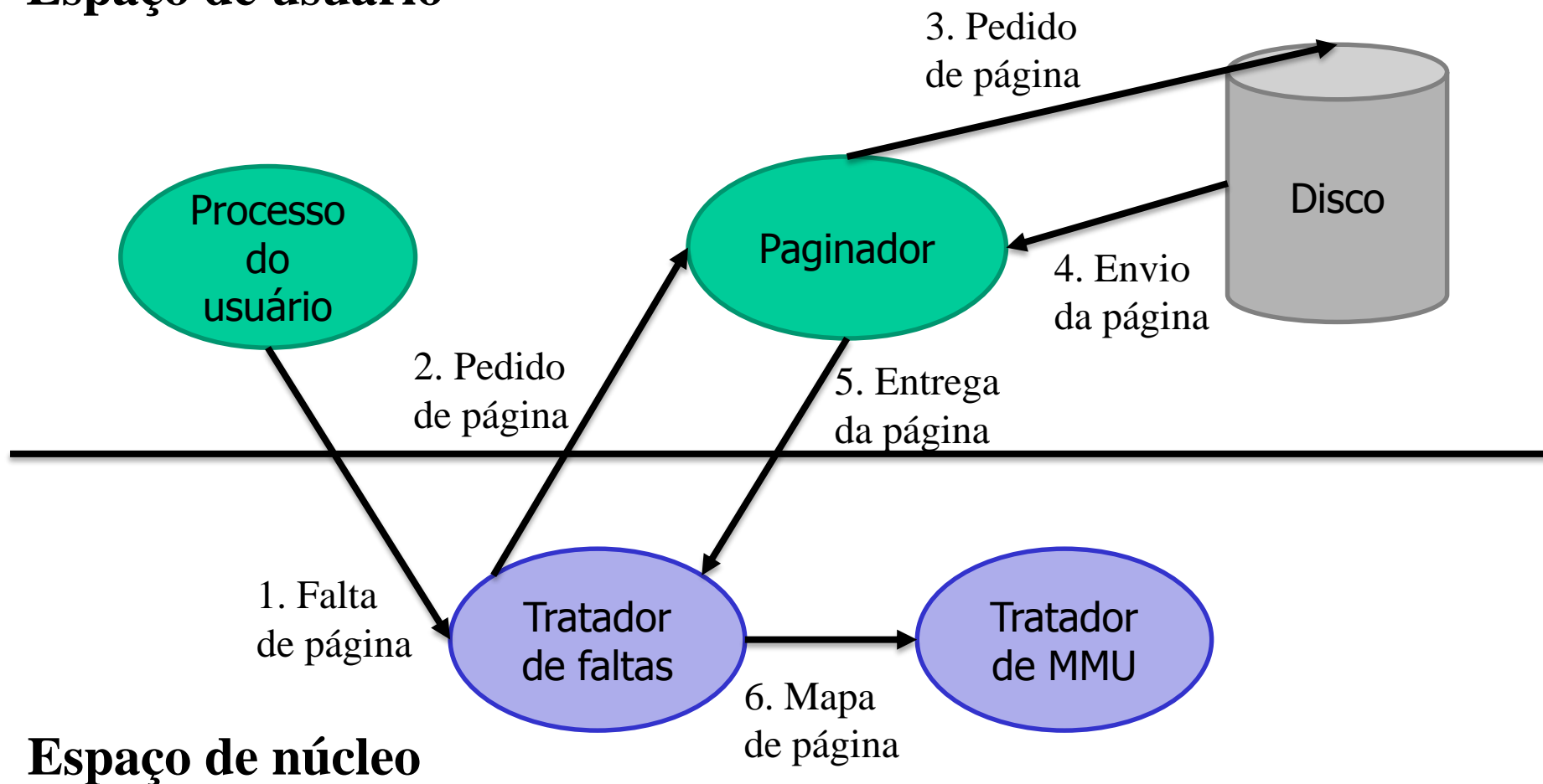
Separação de política e mecanismo



- Tratador de MMU
 - Baixo nível
 - Dependente de hardware
- Tratador de faltas
 - Parte do núcleo
- Paginador
 - Espaço de usuário

Separação de política e mecanismo

Espaço de usuário



- Programa normalmente “cresce” e “encolhe” sua área de memória de forma heterogênea
 - Código estático
 - Pilha e dados variam muito, de formas diferentes
- Memória dividida em setores lógicos
 - Segmentos
- Programador recebe segmentos de memória
 - Endereçamento se dá em duas dimensões
 - Número de segmento
 - Endereço dentro do segmento

Compilação de programas



- Com a segmentação, cada sub-rotina pode ficar em um segmento
 - Chamadas às sub-rotinas são feitas pelo segmento delas
 - Alterar uma única sub-rotina não altera as outras
 - Se rotinas estivessem escritas de forma contígua em uma memória unidimensional, todos os endereços após a rotina mudariam

Proteção com segmentação



- Cada segmento pode ter uma proteção diferente
 - Execução
 - Leitura
 - Escrita
- Segmentos podem ser compartilhados entre processos
 - Bibliotecas
 - Dados compartilhados

Segmentação vs Paginação



- Páginas
 - Tamanho fixo
 - Programador não precisa saber
- Segmentos
 - Tamanho variável
 - Programador deve saber (para poder endereçar)

Implementação



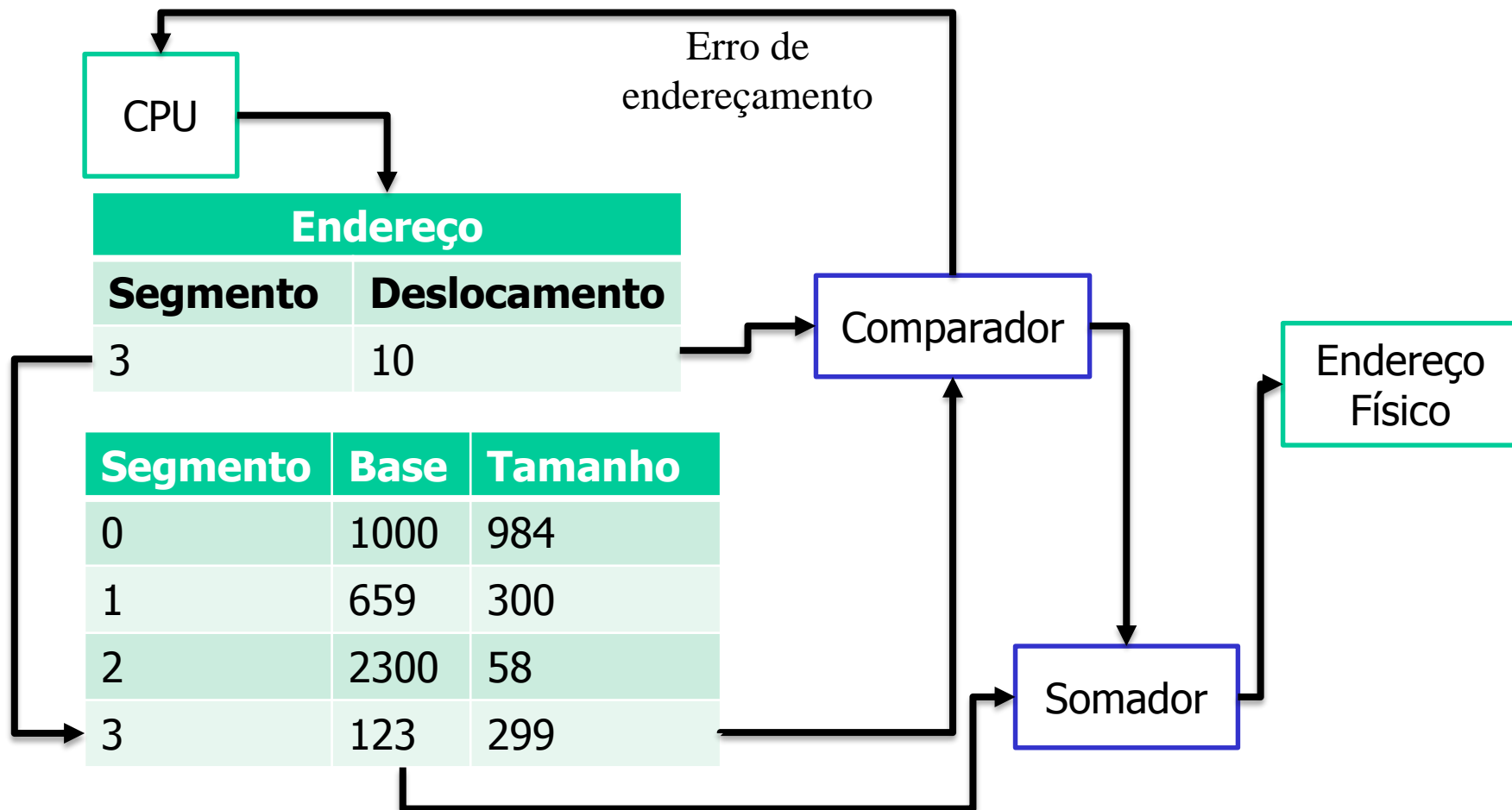
- Segmentos podem ter diferentes tamanhos
- Segmentos podem ser apagados
 - Fragmentação externa
- Cada processo tem uma tabela de segmentos
 - Cada entrada armazena um segmento
 - Início
 - Tamanho
- Segmentos podem ser usados com paginação

Acesso a endereço de memória segmentada



- Programa referencia endereço virtual (#seg, offset)
- Entrada #seg da tabela de segmentos é acessada
 - Retorna o endereço inicial B e o tamanho L do segmento
- Se $\text{offset} < L$
 - Endereço físico = $B + \text{offset}$
- Se não
 - *Segmentation fault*

Endereçamento com segmentação



Segmentação com paginação



- Segmentos nem sempre cabem na memória
 - Ou cabem, mas não ia ser bom mantê-los todos em memória
- Paginação gerencia a memória virtual
- Segmentação gerencia a separação lógica da memória
 - Segmentos possuem suas tabelas de páginas

Implementação



- Processo referencia endereço virtual (#seg, #pag, offset)
- Entrada #seg da tabela de segmentos aponta para a tabela de páginas do segmento
- Entrada #pag da tabela de páginas aponta para um quadro de página
- Endereço físico é o endereço do quadro mais o offset

Segmentação - utilização



- Intel x86-32
 - Dá suporte
 - Windows e Linux não utilizam a segmentação
- Intel x86-64
 - Não dá suporte

Segmentação de Memória

Pedro Cruz

EEL770 – Sistemas Operacionais