

# Sistemas Operacionais

(Visão Geral e Evolução)

Prof. Sérgio Murilo Maciel Fernandes

# Objetivos

- Eficácia para o usuário: criar um sistema de computação utilizável (exemplo: interfaces gráficas – GUIs);
- Operação eficiente do sistema de computação: compartilhamento de recursos;

# Ponto de vista do Usuário

- S.O. para PC – objetiva facilidade de uso, alguma atenção ao desempenho e nenhuma atenção a utilização dos recursos;
- S.O. para mainframe – compromisso entre a utilização dos recursos e a usabilidade individual;
- S.O. embarcado – projetados para operar sem intervenção do usuário

# Ponto de vista do Sistema

- S.O. como gerenciador de recursos – tempo de CPU, espaço de memória, espaço de armazenamento em arquivo, dispositivos de I/O;
- S.O. como programa de controle – gerencia a execução dos programas de usuários e controla os dispositivos de I/O;

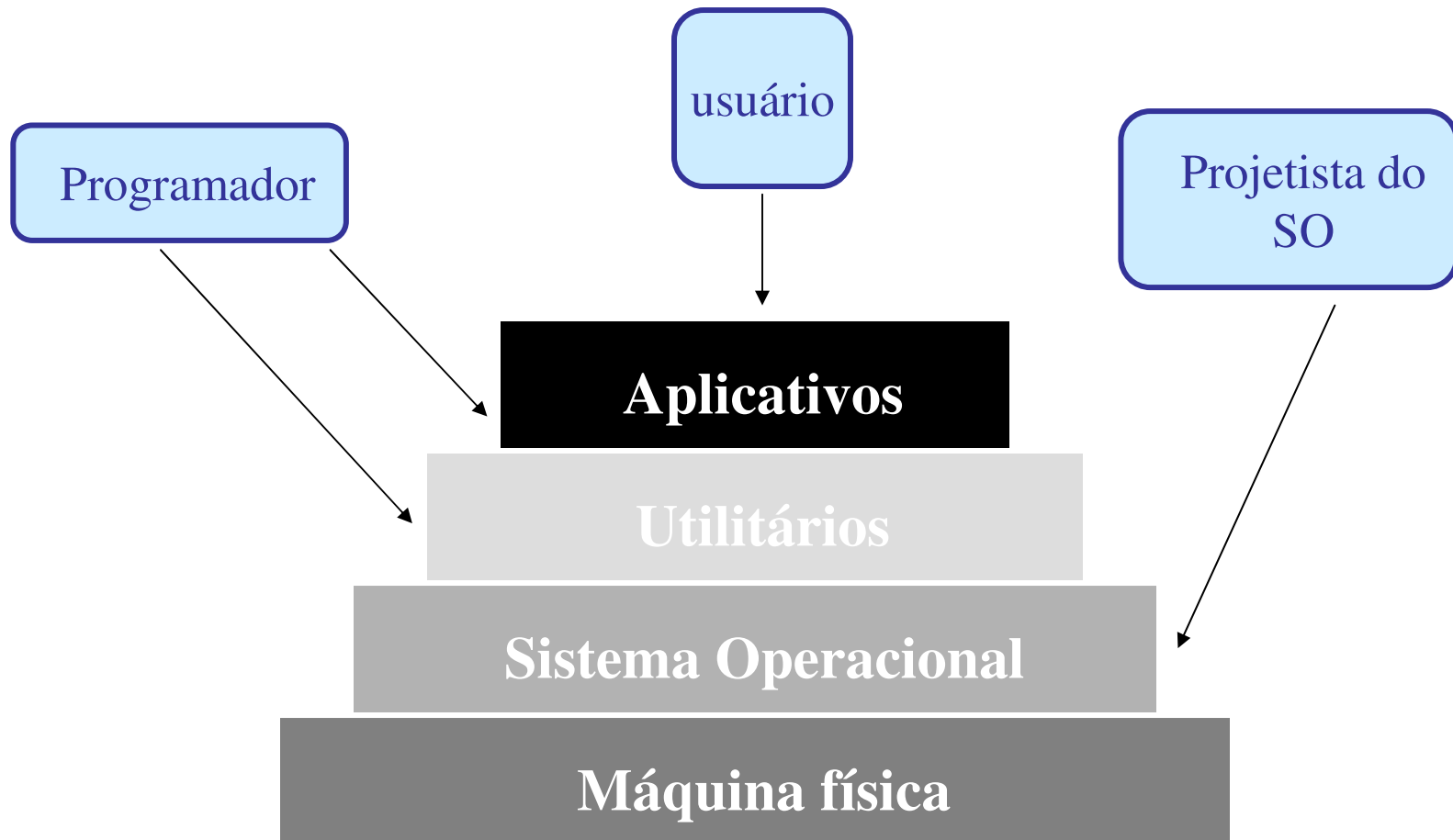
# Definição

- Um programa que controla e coordena o uso do *Hw* pelos diversos programas aplicativos para os diversos usuários;
- Uma interface entre o usuário e o *Hw*;
- Um programa que mascara os detalhes do *Hw*;
- Um programa que cria um ambiente que permite; aos outros programas desempenhar tarefas úteis.

# Definição

- Duas visões
  - Máquina Virtual ou Estendida
  - Gerenciador de Recursos

# Como Máquina Virtual



# Como Máquina Virtual

- Criação de programas
- Execução de programas
- Acesso a dispositivos de I/O
- Acesso controlado a arquivos
- Acesso a memória
- Detecção e correção de erros
- Contabilidade



# Máquina Virtual: Serviços

- Criação de programas
  - SO oferece facilidades: editores e depuradores;
  - tipicamente estes serviços não são parte do SO e sim dos utilitários;
  - contudo, são acessíveis através do SO.

# Como Máquina Virtual

- Execução de programas
  - carregamento do programa em memória;
  - arquivos e dispositivos de I/O devem ser iniciados;
  - outros recursos devem ser preparados;
  - SO gerencia estas ações para o usuário.

# Como Máquina Virtual

- Acesso a dispositivos de I/O
  - Via driver do dispositivo;
  - cada dispositivo tem seu próprio conjunto de instruções ou sinais de controle;
  - SO esconde estas ações e usuário só executa leituras e escritas

# Como Máquina Virtual

- Acesso controlado a arquivos
  - usuário não se preocupa com a natureza do dispositivo de E/S (disco, fita, ...);
  - usuário não se preocupa com formato do arquivo no dispositivo;
  - mecanismos de proteção em caso de múltiplos usuários.

# Como Máquina Virtual

- Acesso ao sistema
  - SO controla acesso ao sistema como um todo e a recursos específicos em particular;
  - i.e., proteção contra acesso não autorizado;
  - i.e., resolução de conflitos em caso de disputa

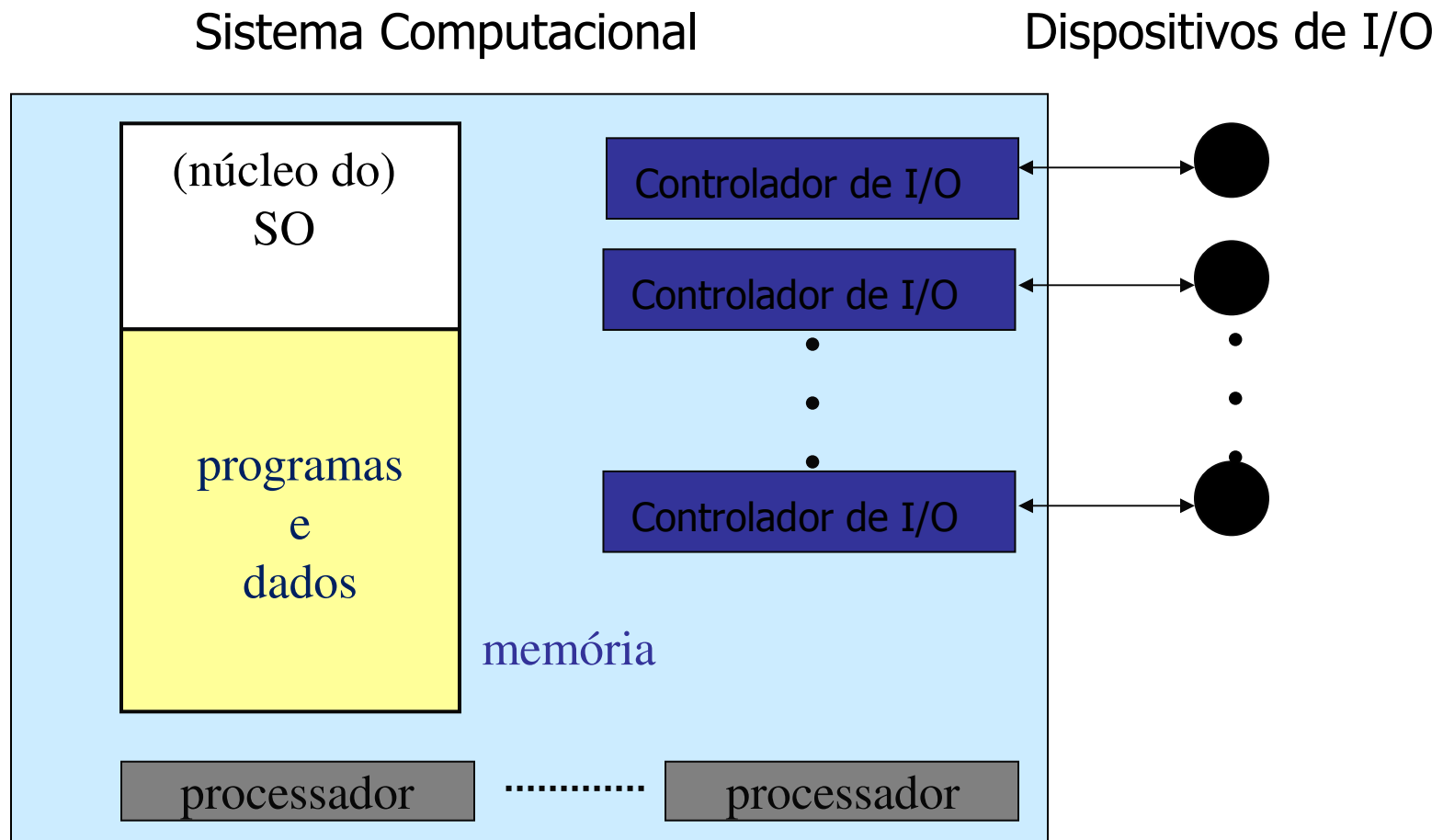
# Como Máquina Virtual

- Detecção e correção de erros
  - erros de *Hw*: memória, dispositivos, ...
  - erros de *Sw*: estouro aritmético, acesso proibido a certas posições de memória;
  - correção da situação com mínimo de impacto no sistema.

# Como Máquina Virtual

- Contabilidade
  - coleta de estatísticas;
  - monitoramento de desempenho;
  - uso: melhoria de desempenho, melhorias futuras;
  - tarifação em um sistema multiusuário.

# Como Gerenciador de Recursos





# Sistema Operacional

- É um **programa**!
- Direciona o processador no uso dos recursos do sistema e sobre o **momento** de executar outros programas;
- S.O. **libera** o processador para que outros programas possam executar.

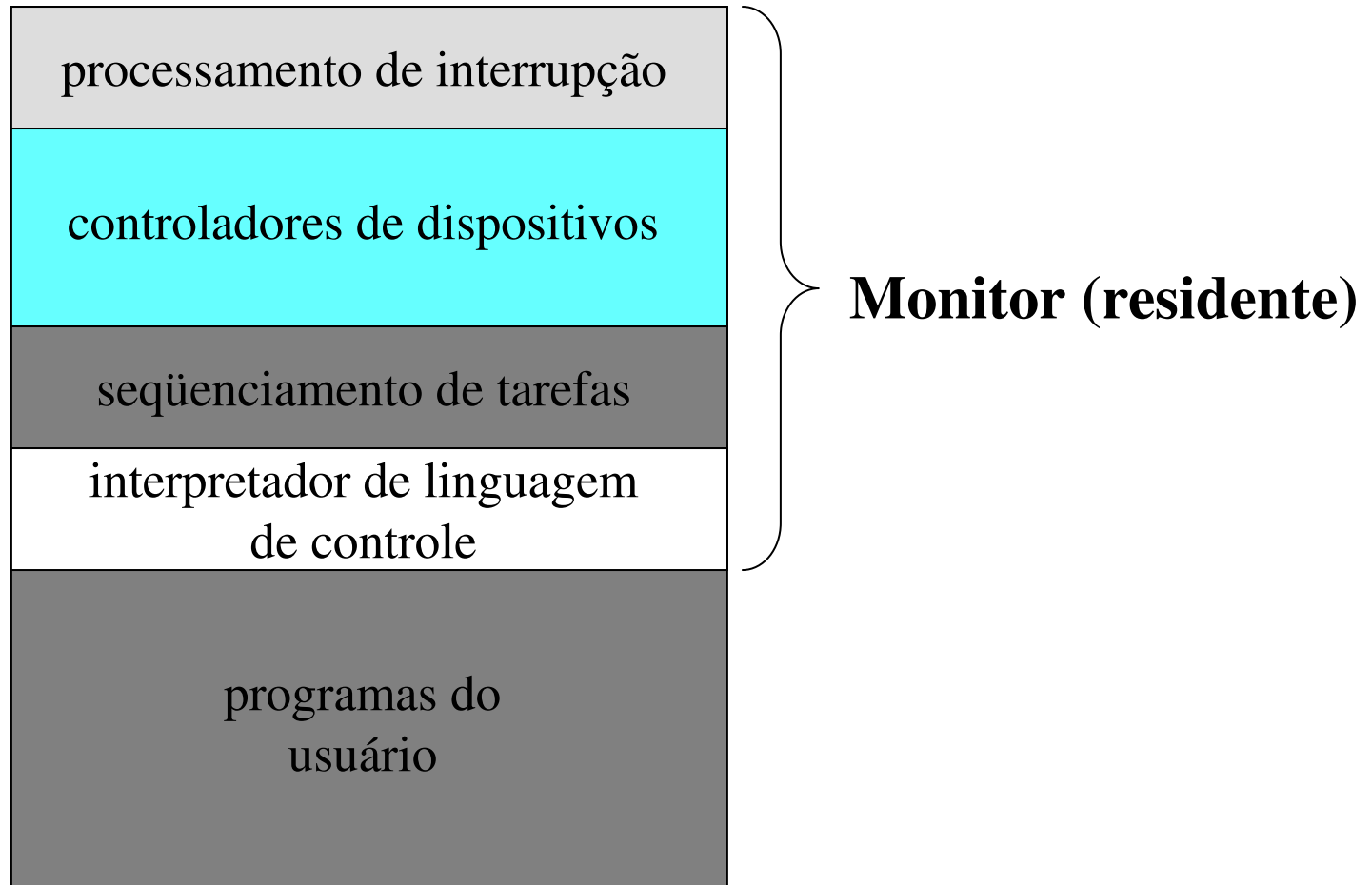
# Evolução dos Sistemas Operacionais

- Um S.O. deve evoluir ao longo do tempo para:
  - receber novos tipos de *hardware* (e.g., novo terminal gráfico);
  - atender novos serviços (e.g., sistema de janelas);
  - reparar defeitos.

# Monitores

- *Software* que controla a execução de outros programas;
- S.O. de lote (*batch*): *jobs* (tarefas) são carregados juntos;
- Monitor é residente em memória principal;
- Utilitários são carregados à medida da necessidade.

# Monitor: Mapa de memória



## JCL: *job control language*

- Tipo especial de linguagem de programação
- Direciona o monitor:
  - que compilador usar
  - que dados usar
  - que dispositivos montar
  - ...

# Características de *Hw* desejáveis

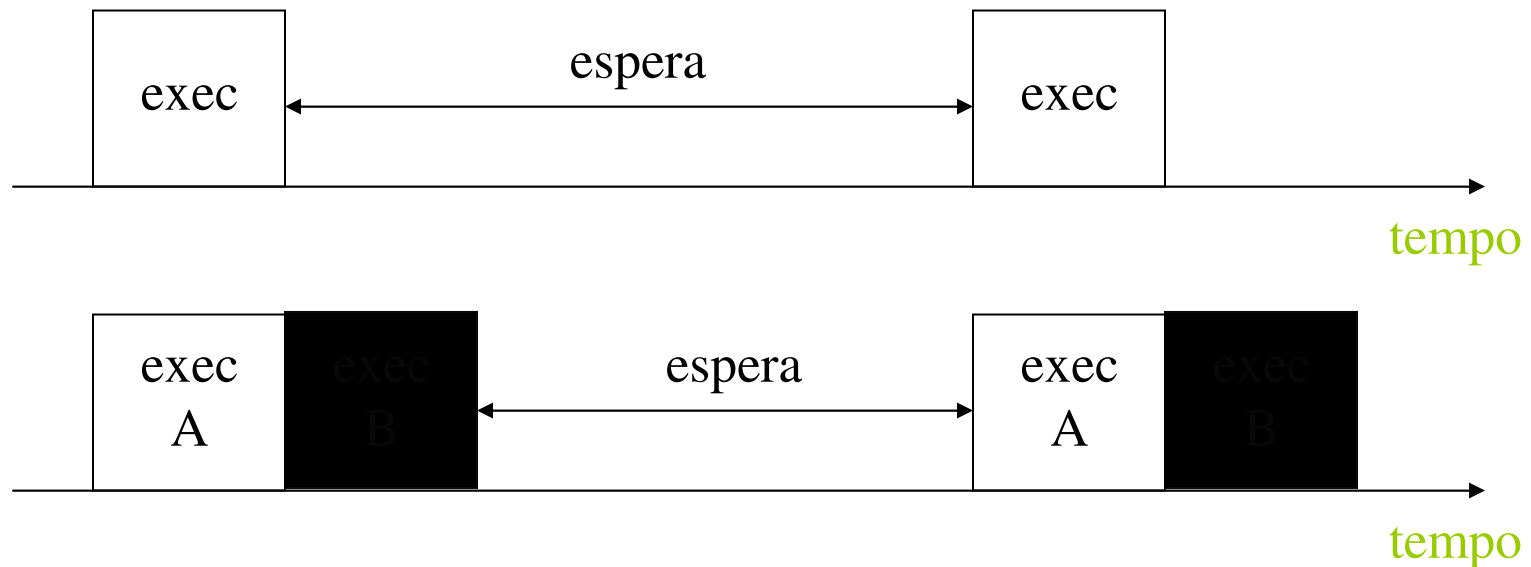
- Proteção de memória
  - não permitir que a área ocupada pelo monitor seja alterada.
- Temporização
  - prevenir um *job* de monopolizar o sistema;
  - ocorrência de interrupção quando o tempo termina.

# Características de *Hw* desejáveis

- Instruções privilegiadas
  - executadas somente pelo monitor, e.g., instruções de E/S;
  - ocorrência de interrupção caso o programa do usuário tente uma dessas instruções.
- Interrupções
  - flexibilidade para controlar programas do usuário.

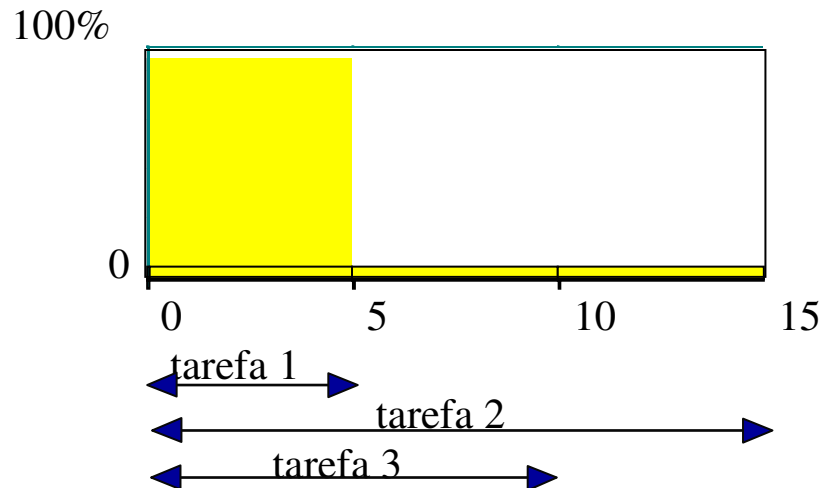
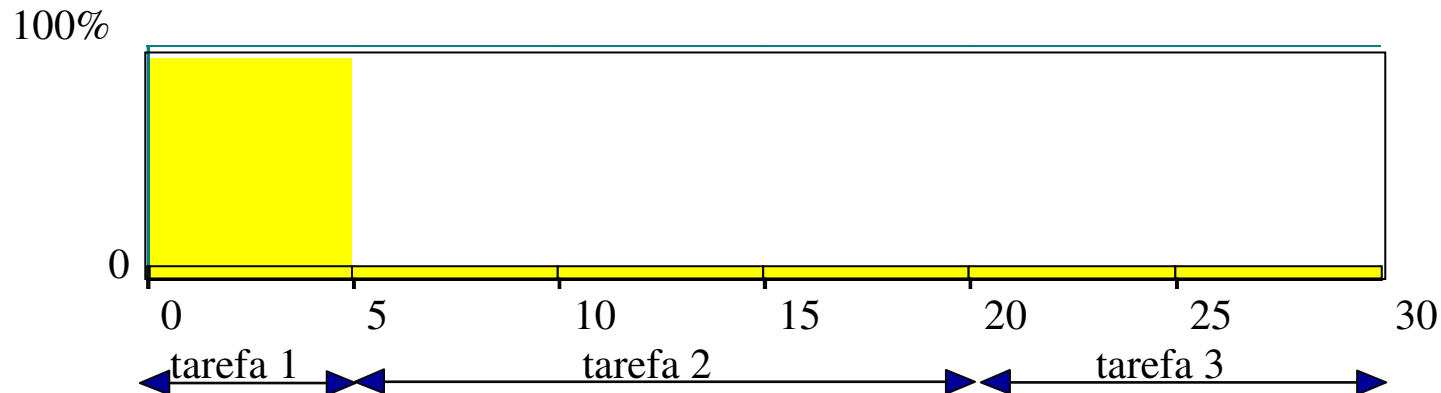
# Multiprogramação

- Permite que o processador execute outro programa enquanto um espera por E/S





# Vejamos....



utilização da UCP:  
monoprogramação ×  
multiprogramação

# Contudo, como tudo tem um preço...

- Necessidade de *hardware* extra, como:
  - E/S por interrupção ou DMA – *Direct memory access*
  - gerenciamento de memória
- Necessidade de *software* extra, como:
  - escalonamento de processos
  - proteção de arquivos
  - sincronização entre processos

# *Time-sharing*

- Uso de multiprogramação para atendimento de tarefas interativas;
- CPU é compartilhada;
- Acesso via terminais.

# Principais tópicos

- Processos;
- Concorrência;
- Escalonamento de processos;
- Gerenciamento de memória;
- Segurança e proteção;
- Formas de estruturação.

# Processos

- Mais geral que programa;
- Consiste em um código executável e seus dados associados, além de um contexto de execução.

# Concorrência

- Principais problemas:
  - sincronização (e.g., perda de sinais);
  - exclusão mútua (e.g., bases de dados);
  - bloqueios (espera infinita).

# Escalonamento de Processos

- Como escolher qual processo ocupará o processador?
- Alguns critérios:
  - justiça
  - diferenciação entre classes
  - vazão máxima
  - minimizar tempo de resposta
- Níveis de escalonamento

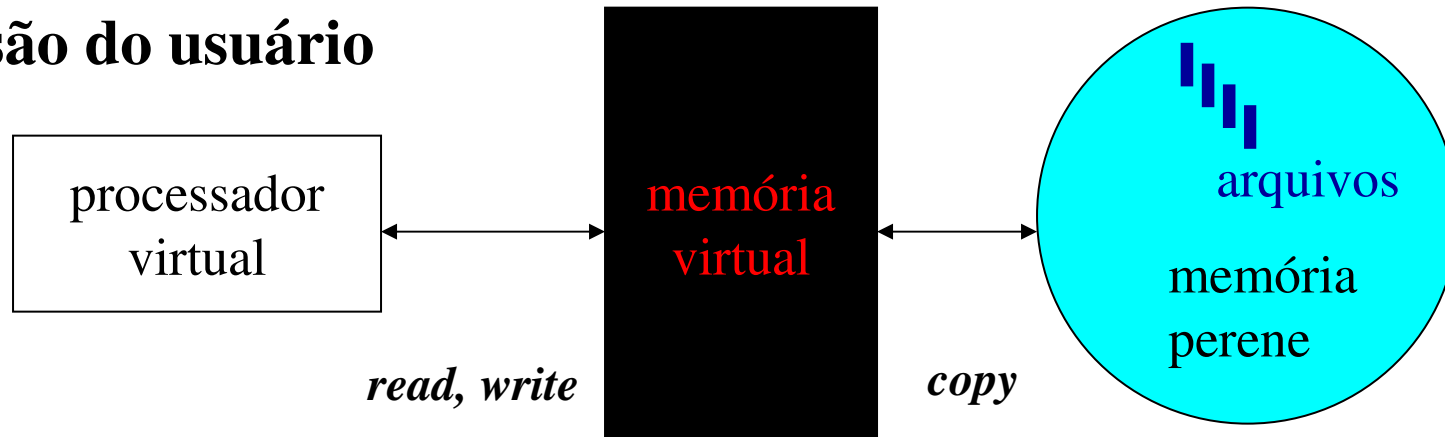
# Gerência de memória

- Requisitos:
  - Hierarquia de memória;
  - Isolação entre processos;
  - Demandas dinâmicas;
  - Proteção e controle de acesso;
  - Armazenamento permanente.
- Solução: memória virtual + sistema de arquivos

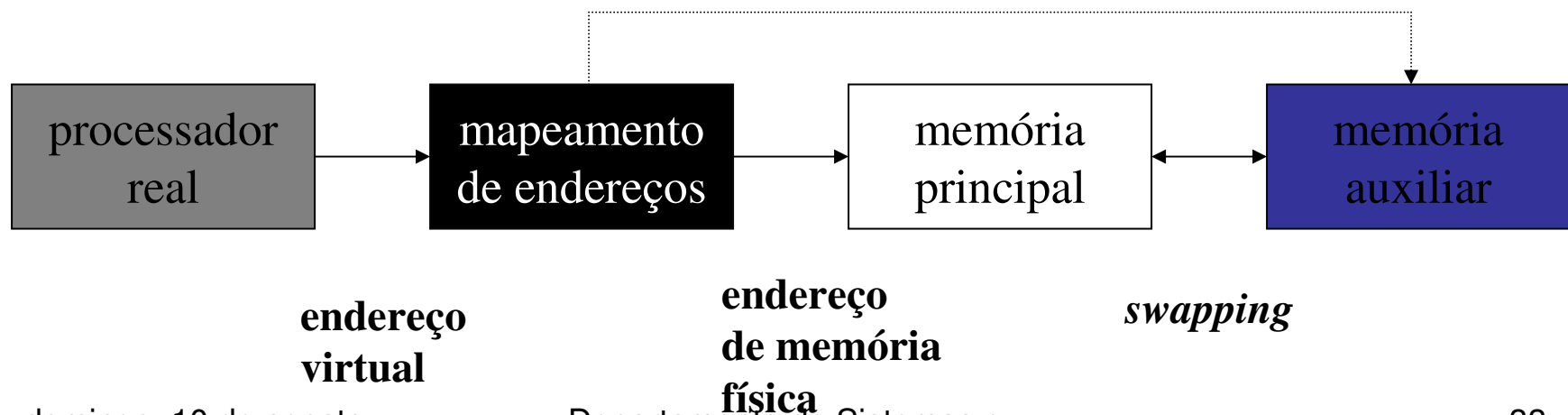


# Memória Virtual

## i) Visão do usuário



## ii) Visão do projetista do S.O.



# Segurança e proteção

- Uma grande preocupação hoje em dia. O que está envolvido?
  - controle de acesso: quem pode acessar sistema e dados?
  - controle de fluxo de informação: quem pode receber o que.
  - certificação: como saber quem é quem?

# Formas de estruturação

- S.Os cada vez mais complexos. Para estruturar, só a programação modular não é suficiente
  - estruturação em níveis;
  - arquitetura em micro-núcleo;
  - *threads* e *multithreads*.

# Outras Formas.....

- Multiprocessamento simétrico
  - cada processador executa cópia do S.O.
- S.Os distribuídos
  - fornece a ilusão de uma única memória principal
- Sistemas móveis

# Outros Requisitos.....

- Sistemas de tempo real (TR)
  - normalmente usados em aplicações dedicadas
  - requisitos temporais bem definidos
  - sistemas TR críticos
    - vale a pena usar memória virtual?
  - sistemas TR não-críticos
- Consumo de energia

# Onde obter informação complementar?

- *Fundamentos de Sistemas Operacionais*,  
A. Silberschatz e P.B. Galvin, Addison-  
Wesley
- *Sistemas Operacionais Modernos*,  
A.S. Tanenbaum, Prentice Hall