Sistemas Operacionais

Sumário: UNIDADE 1: Estrutura Sistemas Operacionais

→ Gerência de Processos

→ Gerência de Memória Principal

→ Gerência de Memória Secundária

→ Gerência de Sistemas de I/O

→ Gerência de Arquivos

→ Sistema de Proteção

→ Sistema Interpretador de Comandos

→ Serviços do S.O.

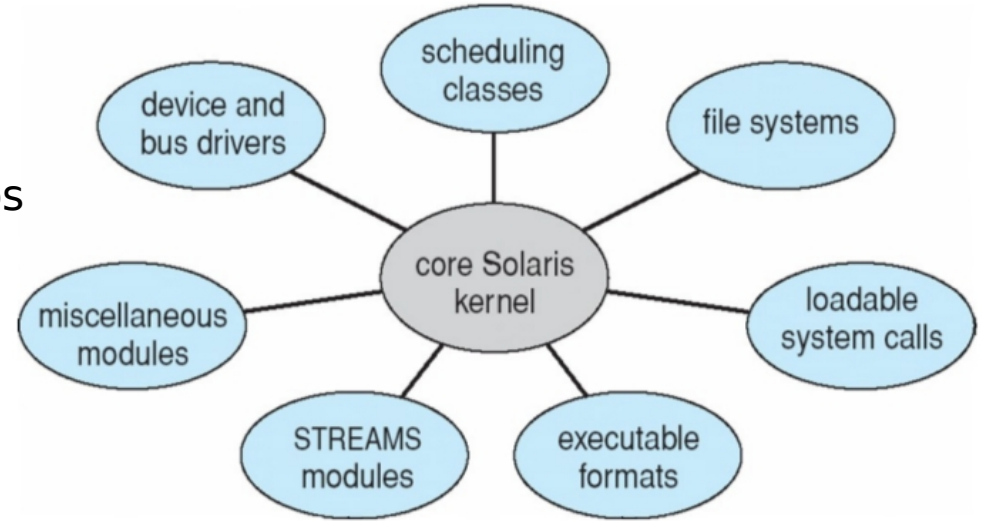
→ Chamadas ao Sistema

→ Programas de Sistema

→ Estrutura do Sistema

Abordagem Simples, Camadas e Módulos

→ Máquinas Virtuais



PUC Minas

Gerência de Processos

- Um processo é um programa em execução que necessita de recursos como tempo de CPU, memória, arquivos e dispositivos de I/O para executar sua tarefa.

- O SO É responsável pelas atividades relacionadas à gerência de processos:

  - Criação e exclusão de processos
  - Suspensão e reativação dos processos
  - provisionamento de mecanismos para:

    sincronização de processos e comunicação entre processos

PUC Minas

Gerência de Memória Principal

- Memória é uma grande arranjo de palavras ou bytes, cada um com seu próprio endereço. É um repositório de dados de acesso rápido, compartilhado pela CPU e dispositivos de I/O.

- O SO é responsável pelas seguintes atividades relacionadas à gerência de memória:
  - Manter registros de quais partes da memória estão sendo utilizadas e por quem
  - Decidir quais processos carregar quando existe memória disponível
  - Alocar e liberar espaço de memória quando preciso

PUC Minas

Gerência de Memória Secundária

Uma vez que a memória principal volátil e pequena para acomodar todos os programas e dados de forma permanente, o sistema de computação deve prover memória secundária.

A memória dos sistemas modernos utilizam discos como principal meio de armazenamento on-line.

O SO. é responsável pelas seguintes atividades relacionadas à gerência de discos:

Gerência de espaço Livre

- Alocação de espaço de memória
- Escalonamento de disco

PUC Minas

Gerência de Sistemas de I/O

O sistema de I/O consiste em:

Um sistema de *buffer* (acumulador) *cache*.

Uma interface geral de *driver* de dispositivo

Driver para dispositivos de hardware específicos

PUC Minas

Gerência de Arquivos

Um arquivo é uma coleção de informações relacionadas definido pelo seu criador. Podem representar programas e dados:

O SO é responsável pelas seguintes atividades relacionadas à gerência de arquivos:

- Criação e exclusão de arquivos
- Criação e exclusão de diretórios
- Suporte a primitivas para manipulação de arquivos e diretórios
- Mapeamento de arquivos em Memória Secundária

PUC Minas

Sistema de Proteção

Proteção refere-se a mecanismos para controlar o acesso de programas, processos e usuários aos recursos do sistema e de seus usuários.

O mecanismo de proteção deve:

- Distinguir entre uso autorizado e não autorizado
- Especificar o controle a ser imposto
- Prover meios para seu cumprimento

PUC Minas

Sistema Interpretador de Comandos

Vários comandos são sados ao SO por instruções de controle que tratam de:

- criação e gerência de processos
- gerência de I/O
- gerência de memória secundária
- gerência de memória primária
- acesso ao sistema de arquivos
- proteção
- acesso à rede

O programa que lê e interpreta instruções de controle é chamado de interpretador de linha de comando ou ***shell***.

Sua função é receber e executar o próximo comando do usuário.

PUC Minas

Serviços do Sistema Operacional

Execução de programas – capacidade de carregar um programa na memória e executá-lo.

Operações de I/O – uma vez os programas usuários não podem executar operações de I/O diretamente, o SO.

Manipulação de Sistemas de Arquivos – capacidade de ler, escrever, criar e remover arquivos.

Comunicação – troca de informações entre processos executando na mesma máquina ou em sistemas conectados. Implementado através de memória compartilhada ou passagem de mensagens.

Detecção de erros – garantir a computação correta, detectando erros na CPU, hardware de memória, dispositivos de I°O e programas de usuários.

PUC Minas

Chamadas ao Sistema

Chamadas ao sistema *(System calls)* oferecem uma interface entre um processo e o sistema operacional.

Normalmente disponíveis como instruções ***assembly***

Linguagem podem substituir estas instruções por sub-rotinas ou chamadas diretas

Geralmente, 3 métodos são utilizados para passar parâmetros ao SO:

- Colocá-los em registradores

- Guardá-los em tabelas na memória, cujo endereço fica em um registrador

- Empilhar os parâmetros na pilha

PUC Minas

Chamadas ao Sistema

Windows (DOS) e Linux (Unix)

	Windows	Unix
Controle de processo	CreateProcess( ) ExitProcess( ) WaitForSingleObject( )	fork( ) exit( ) wait( )
Gerenciamento de Arquivos	CreateFile( ) ReadFile( ) WriteFile( ) Close( )	open( ) read( ) write( ) close( )
Gerenciamento de dispositivos	SetConsoleMode( ) ReadConsole( ) WiteConsole( )	Ioctl( ) Read( ) Write( )
Manutenção de informações	GetCurrentProcessID( ) SetTimer( ) Sleep( )	getpid( ) alarm( ) sleep( )
Comunicação	CreatePipe( ) CreteFilemapping( ) MapViewOfFIle( )	pipe( ) shmget( ) nmap( )
Proteção	SetFileSecurity( ) InitliazeSecurityDescriptor( ) SetSecurityDescriptorGroup( )	chmod( ) umask( ) chown( )

PUC Minas

Programas de Sistema

Programas de sistema oferecem um ambiente prático para o desenvolvimento e execução de programas.

Podem ser divididos:

- Gerência de arquivos
- Informação de *status*
- Modificadores de arquivos
- Suporte a linguagem de programação
- Carregamento e execução de programas
- Comunicação
- Aplicativos

PUC Minas

Estrutura do Sistema

Abordagem Simples

MS-DOS – Escrito para dar o máximo de funcionalidade no mínimo espaço

- Não divido em módulos
- Embora o MS-DOS tenha alguma estrutura, sua interface e níveis de funcionalidades não são bem separados.
- Escrito como uma coleção de rotinas, sendo que cada rotina pode fazer chamadas a qualquer outra.

Exemplos: O DOS e os primeiros sistemas operacionais baseados no Unix.

PUC Minas

Estrutura do Sistema

Abordagem Simples

UNIX – dividido em 2 partes separáveis:

Programas de Sistemas

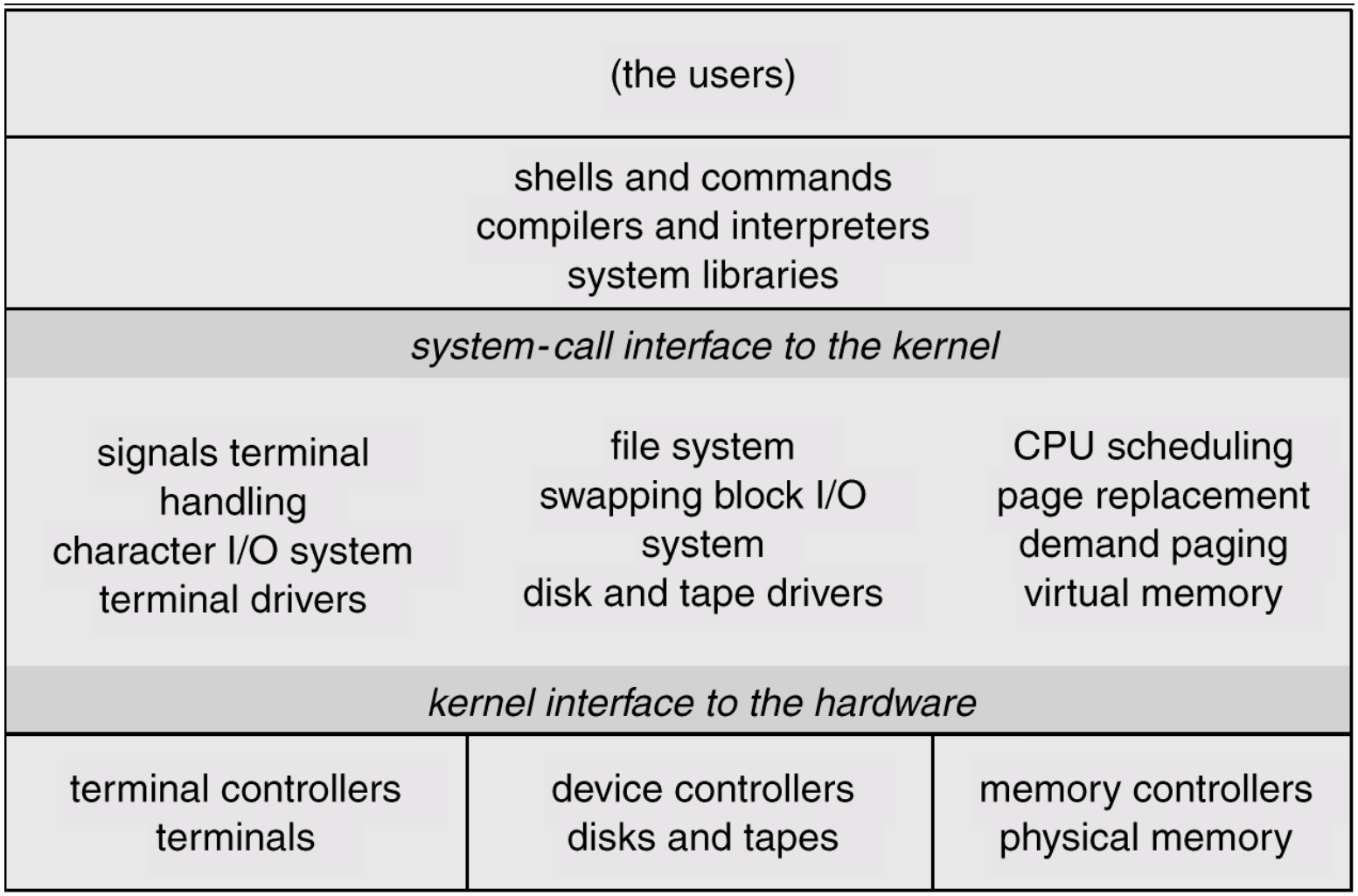
O **Kernel**

Consiste em tudo o que vem abaixo da interface de chamadas ao sistema e acima do hardware

Provê gerência de sistemas de arquivos, escalonamento de CPU, gerência de memória e outras funções.

PUC Minas

Estrutura do UNIX



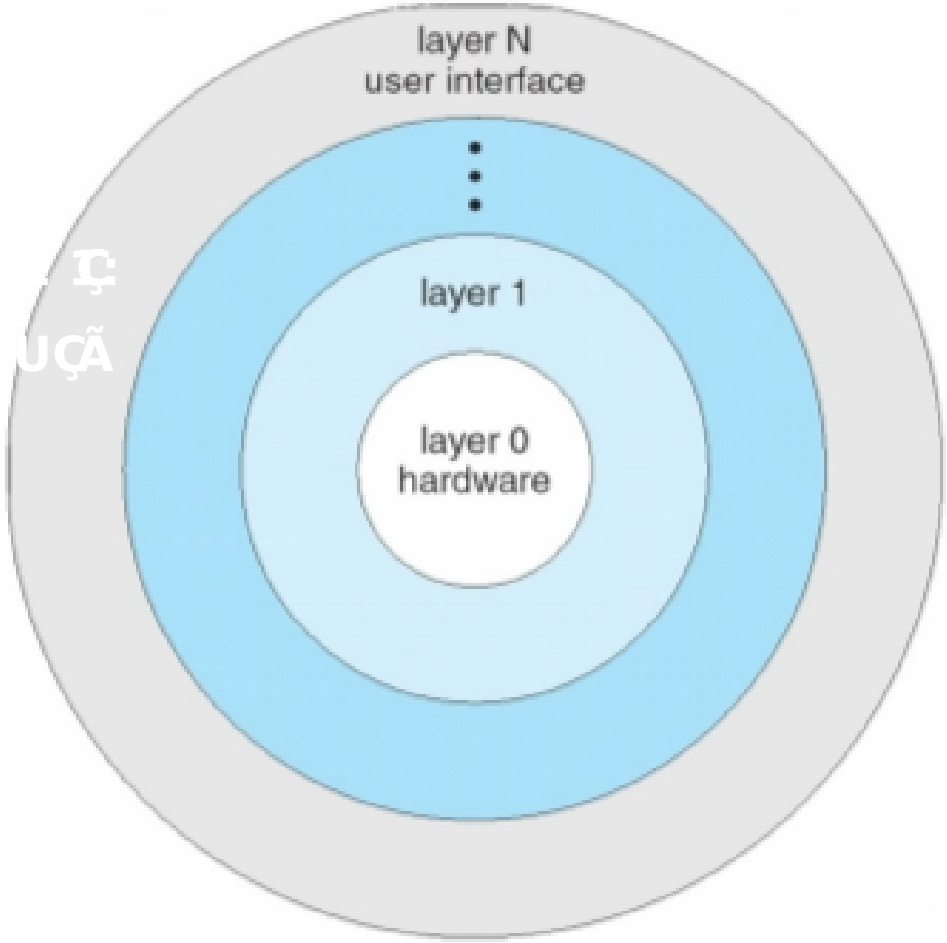
PUC Minas

Estrutura do Sistema

Abordagem em Camadas

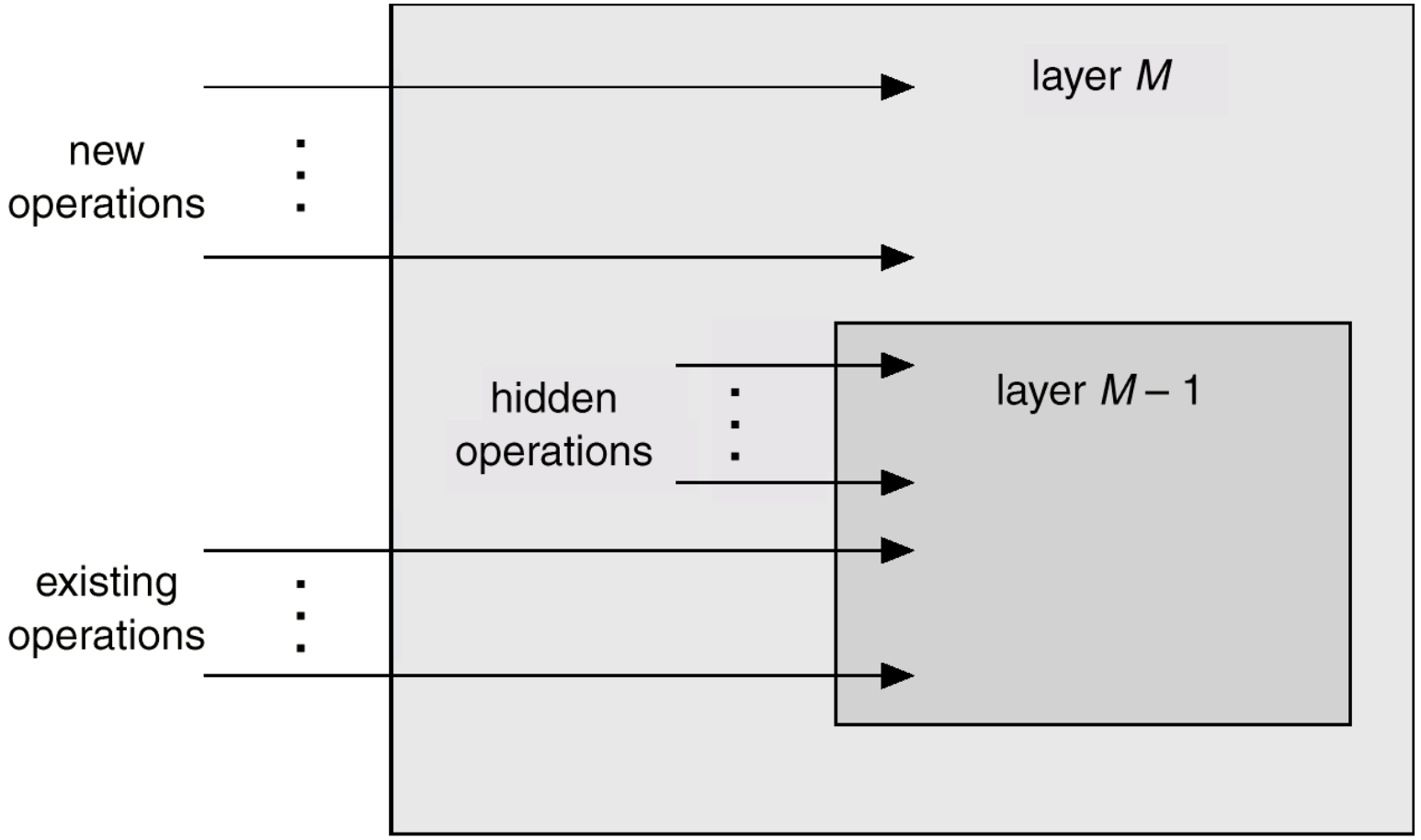
O SO é dividido em um número de camadas (níveis). A camadas de baixo (nível 0) é o hadware; a mais alta (nível N) é a interface com o usuário.

Cada camada usa funções e serviços apenas das camadas inferiores.



PUC Minas

Uma camada do SO.



PUC Minas

Microkernels

A medida em que o Unix se expandiu, o *kernel* tornou-se grande e difícil de gerenciar.

Microkernels: modularização do kernel (CMU).
Componentes não essenciais são implementados como programas do sistema

Principal função do kernel é fazer a comunicação entre os programas clientes e os serviços disponíveis

Vantagem: facilidade de expandir o SO. - novos serviços são adicionados com a mínima alteração do *kernel*.

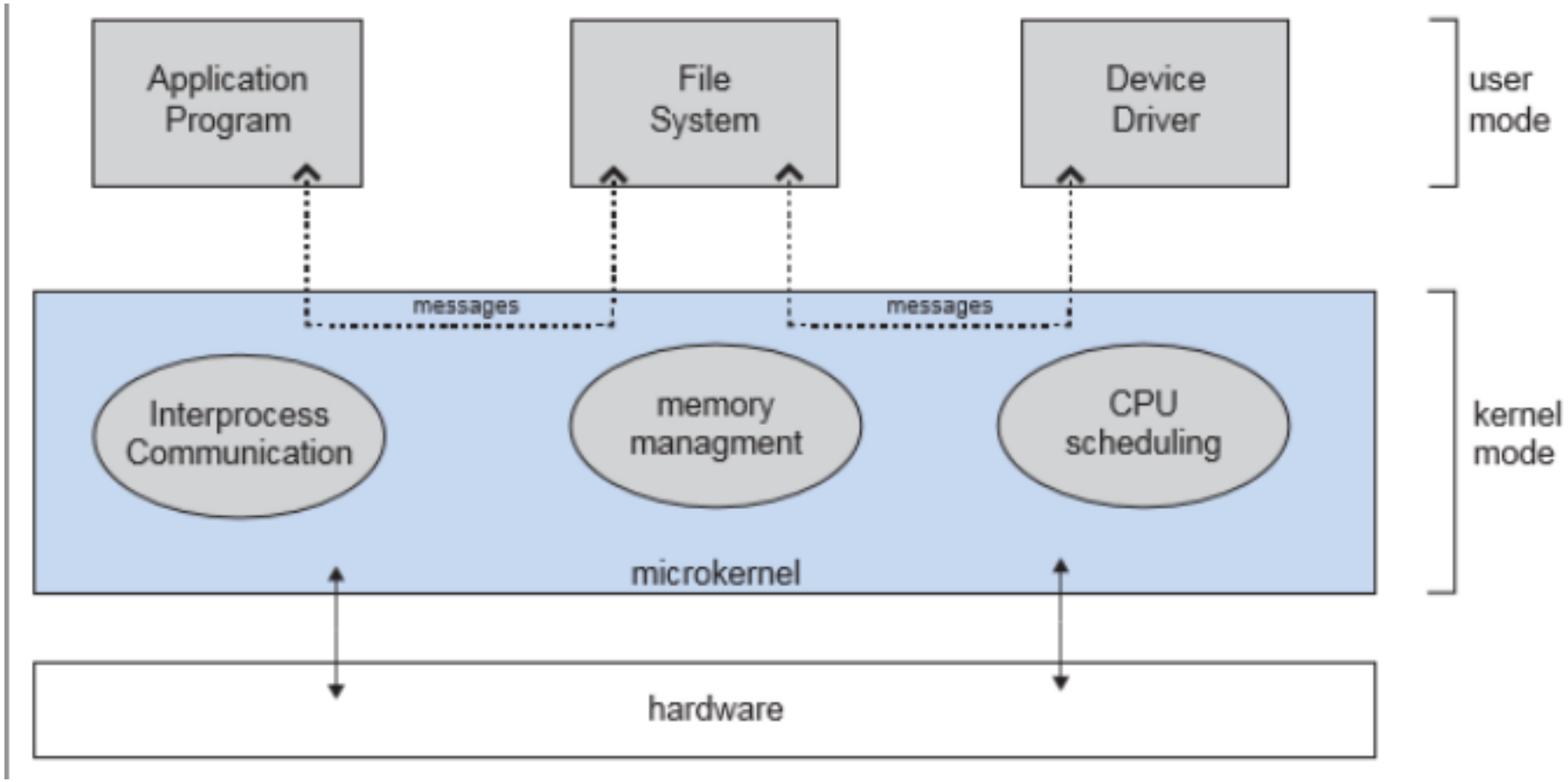
Desvantagem: *Overhead* do espaço do usuário para espaço do *kernel*.

Sistemas baseados em *microkernels*: Apple MacOS, Windows NT, Unix Digital.

PUC Minas

Microkernels

Diagrama da Estrutura do Microkernel



PUC Minas

Estrutura em Módulos

Abordagem em módulos

A maioria dos sistemas operacionais modernos implementa módulos em *kernel*.

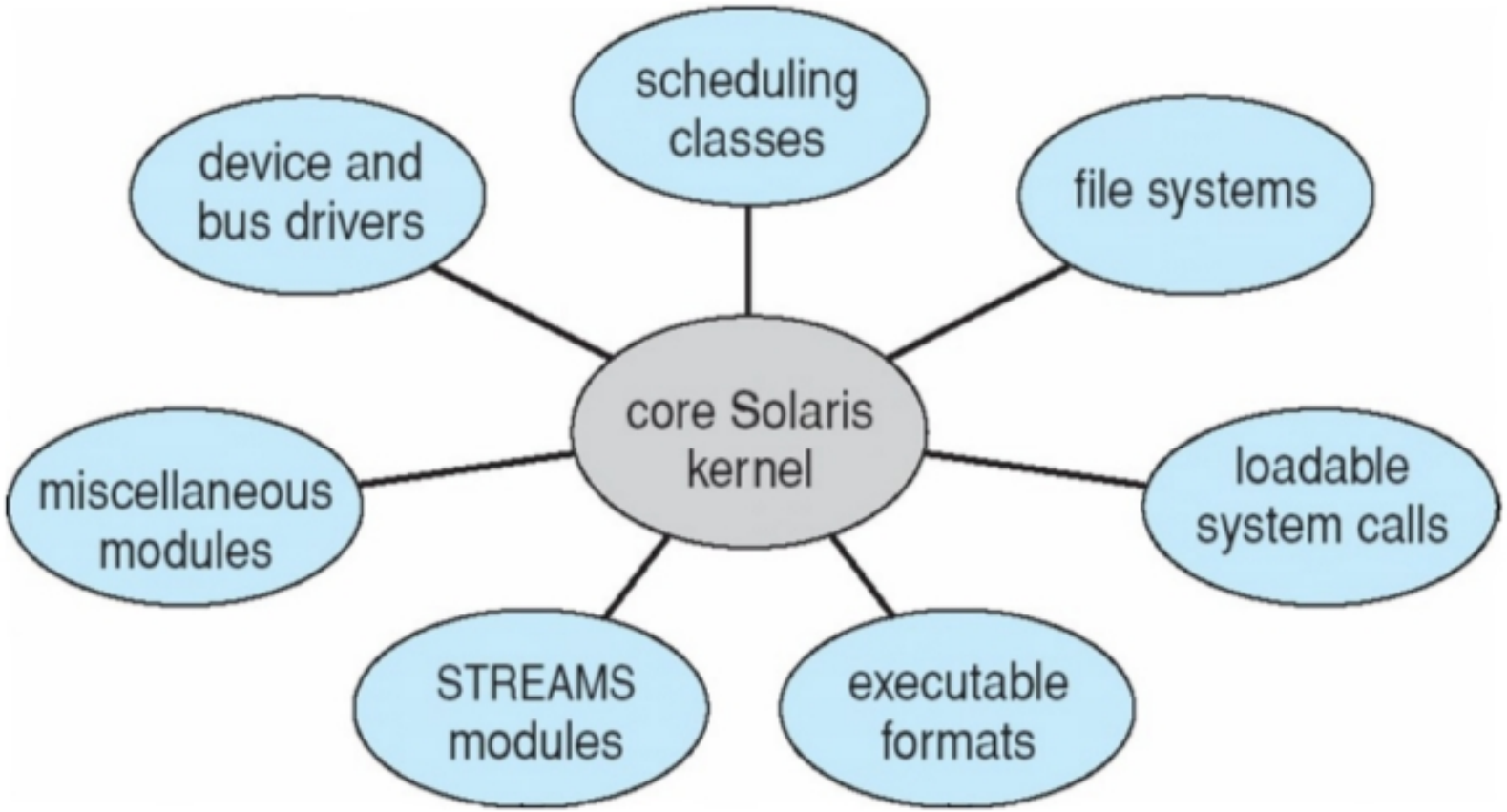
- Usa abordagem orientada a objeto
- Cada componente principal é separado
- cada componente conversa com outros através de interfaces conhecidas
- Cada componente é carregado conforme necessário dentro do *kernel*

Normalmente, semelhante à estrutura em camadas, no entanto é mais flexível.

PUC Minas

Estrutura em Módulos

Diagrama da Abordagem em módulos



PUC Minas

MAQUINAS VIRTUAIS

Conceitos

Uma máquina virtual se baseia na implementação de sistemas por camadas. Ela trata o hardware e o *kernel* do SO. Como se fossem um só hardware (IBM).

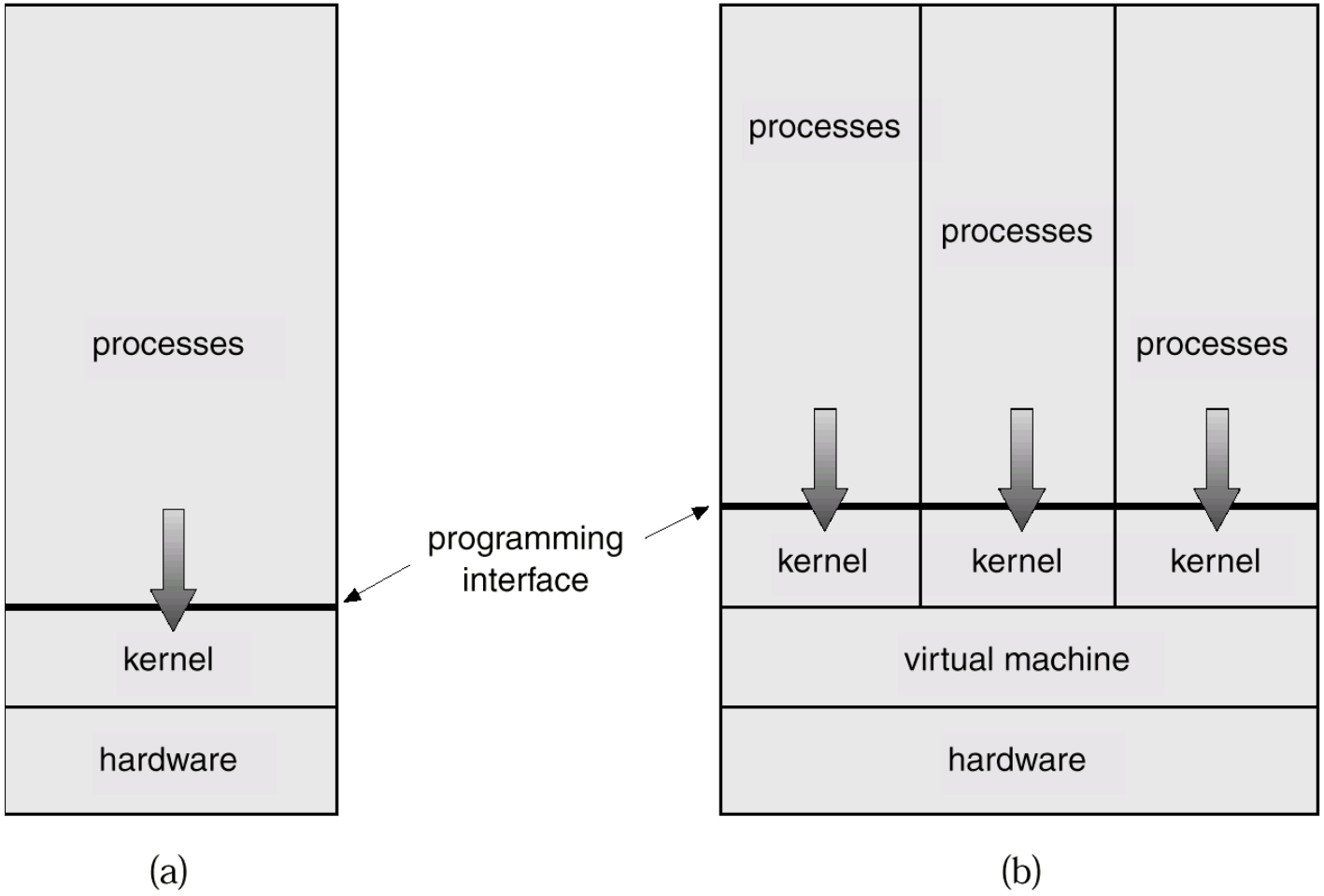
Uma máquina virtual oferece uma interface idêntica ao hardware.

O SO Cria uma “ilusão” de múltiplos processos, cada qual executando em seu próprio processador, com sua própria memória (virtual). Os recursos da máquina física são compartilhados para criar as máquinas.

Maquinas Virtuais

Modelos de Sistemas Física vs VM (direto no Hardware)

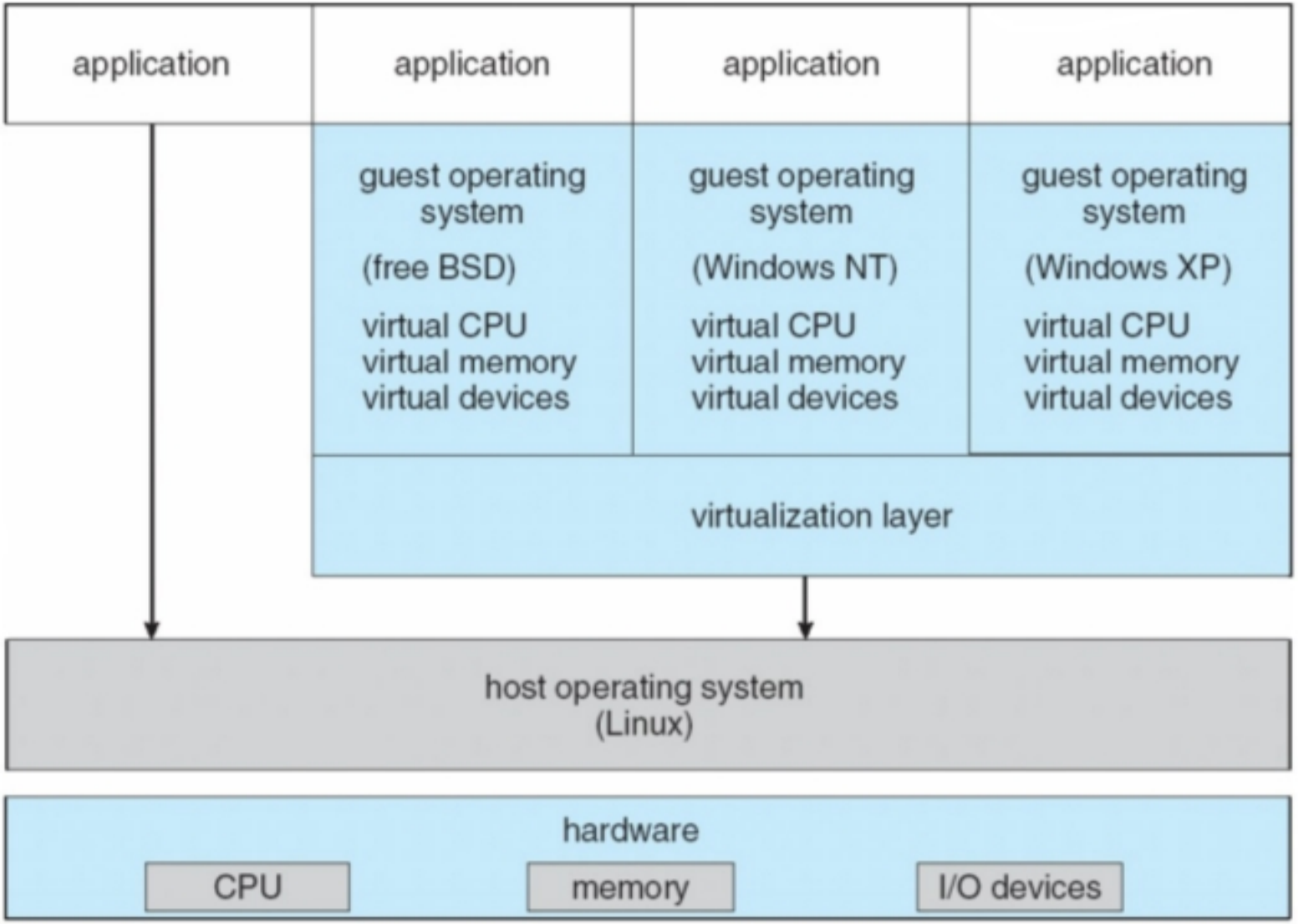
- Non-virtual Machine
- Virtual Machine



PUC Minas

Arquitetura VM

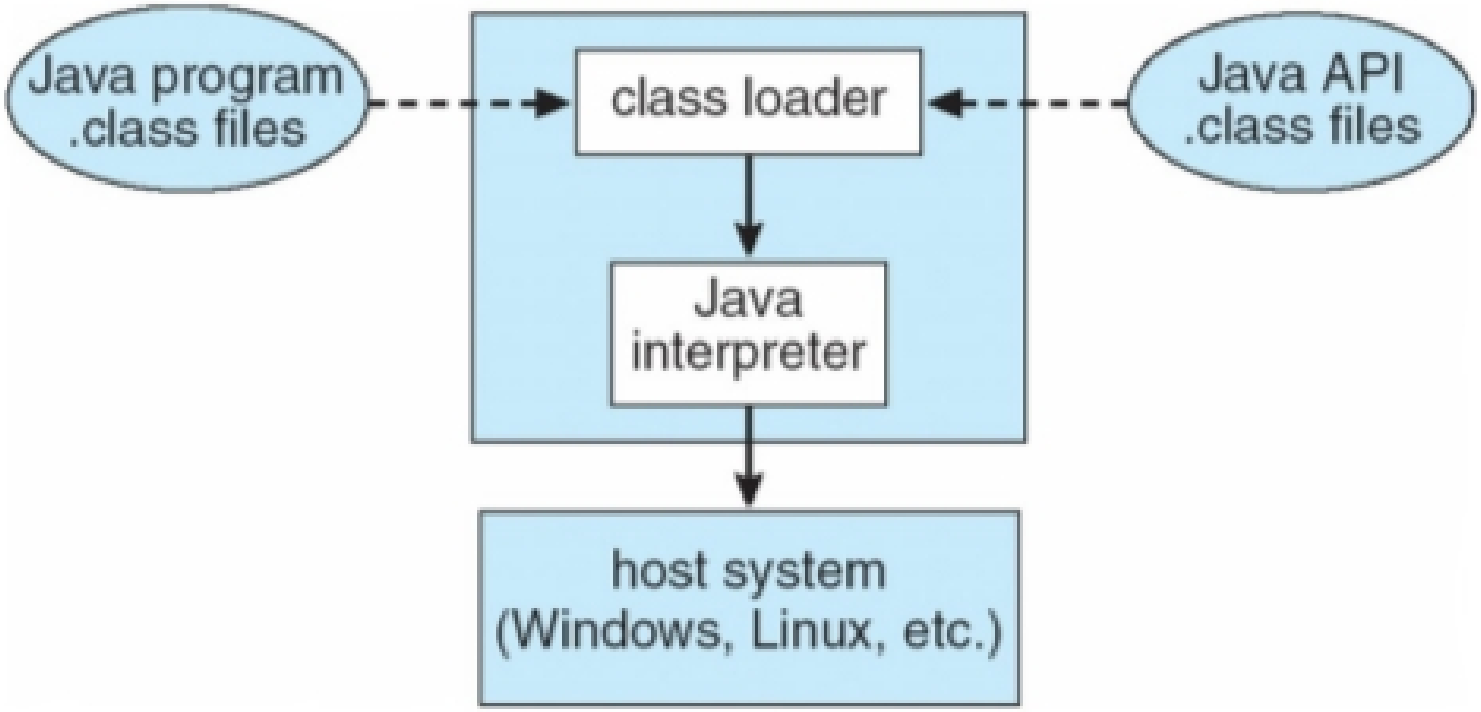
Arquitetura de VMWare (sobre SO)



PUC Minas

Máquina Virtual Java

Arquitetura da JVM



O uso da VM Java permite que um mesmo programa java (*bytecode*) execute em diferentes ambientes computacionais.

PUC Minas

Maquinas Virtuais

VANTAGENS E DESVANTANTES

O Conceito de VM oferece completa proteção dos recursos do sistema, já que cada VM é isolada das demais. Isto, no entanto, impede o compartilhamento direto de recursos.

Um sistema de VMs é ideal para desenvolvimento e pesquisa em SO Já que desenvolver em uma VM não afeta as demais.

O conceito de VM é difícil de implementar devido ao esforço necessário para oferecer uma duplicada exata da maquina original.

Vantagens

- Executar Múltiplos Sistemas Operacionais Simultaneamente
- Portabilidade para Sistemas Legados embutidos
- Embutir soluções de servidores de forma exclusiva em VM
- Recuperação de Desastres
- Economia na infraestrutura (hardware e eletricidade)

DESVANTAGENS

- Sobrecarga na máquina física afeta todas as máquinas virtuais
- A emulação em si é um programa em execução, portanto, consome recursos do hardware.

Referências

Tanenbaum A. S; Woodhull A. S. **Sistemas Operacionais modernos.** São Paulo: Person Prentice Hall, 2009. XVI, 633 p. ISBN 9788576052371

SILBERSCHATZ, Abraham; GALVIN, Peter B.; GAGNE, Greg. **Fundamentos de sistemas operacionais**: princípios básicos. Rio de Janeiro, RJ: LTC, 2013. xvi, 432 p. ISBN 9788521622055