Universidade Federal de Uberlândia
Faculdade de Computação
Sistemas Operacionais

Arquitetura de Computadores: uma breve revisão

Prof. Dr. Marcelo Zanchetta do Nascimento

Sumário

- Estrutura de Sistema Computacional
  - Sistema de Boot
  - CPU
  - Interrupções
  - DMA
- Hierarquia de Armazenamento
- Proteção de Hardware
- Virtualização
- Arquitetura
- Leituras Sugeridas

Estrutura de Sistema de Computação

Arquitetura de Sistema de Computação

- Uma ou mais CPUs e dispositivos de controle conectados por meio de barramentos para fornecer acesso a memória compartilhada.

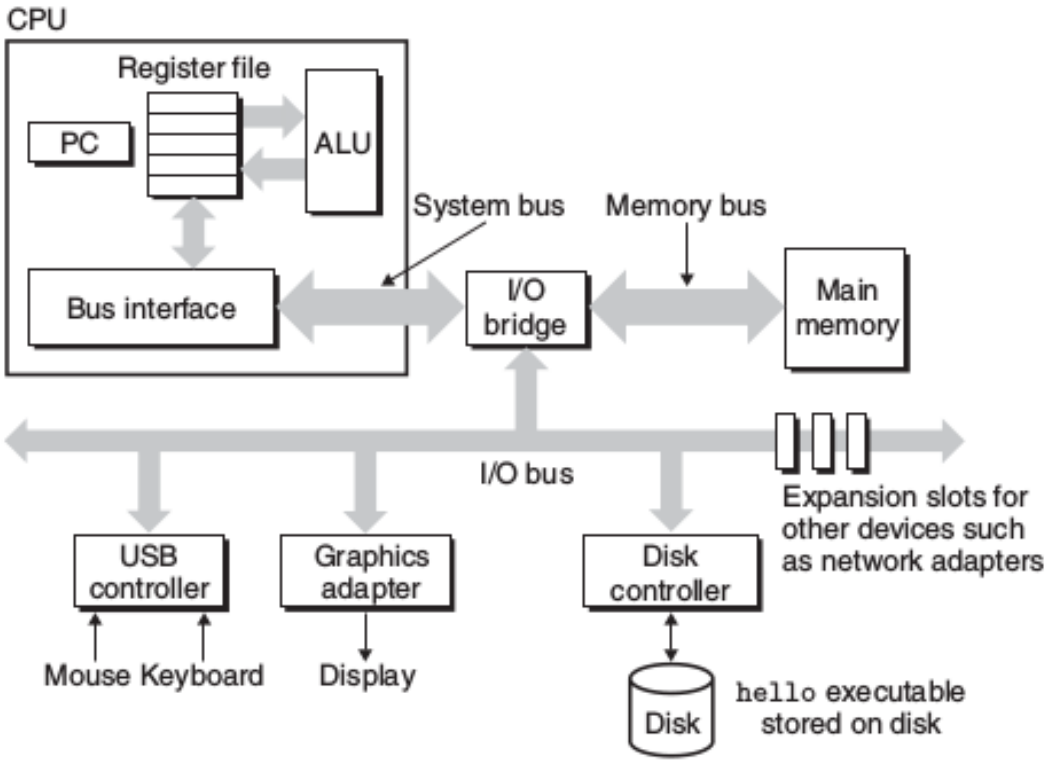


Figura 1: Exemplo de uma Arquitetura Computacional

Sistema de Boot

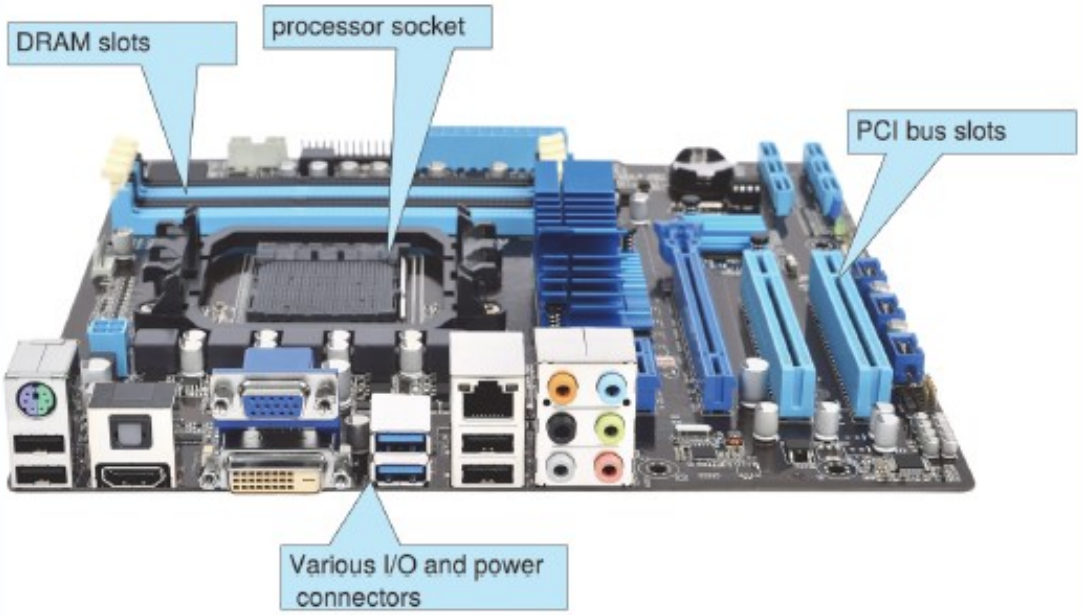
Programa de bootstrap

- Armazenado em uma ROM ou EPROM, geralmente, conhecido como **firmware;**
- Inicializa os dispositivos de um sistema;
- Um software de **bootstrap loader**, **GRUB**, permite a seleção do kernel de múltiplos discos, versões, etc;
- O kernel é carregado e o sistema então é **executado.**

Estrutura de Sistema de Computação

PC MOTHERBOARD

Consider the desktop PC motherboard with a processor socket shown below:



This board is a fully functioning computer, once its slots are populated. It consists of a processor socket containing a CPU, DRAM sockets, PCIe bus slots, and I/O connectors of various types. Even the lowest-cost general-purpose CPU contains multiple cores. Some motherboards contain multiple processor sockets. More advanced computers allow more than one system board, creating NUMA systems.

Estrutura de Sistema de Computação

Informações da motherboard (placa mãe):

- `$ lspci |more`
- `$ sudo dmidecode -t 2`
- `$ sudo lshw`
- `$ hwloc-ls`

Informações da CPU:

- `$ lscpu`
- `$ cat /proc/cpuinfo`

Estrutura de Sistema de Computação

- Um programa de usuário faz um ***system call***, o qual ativa trap (exceção) e invoca o SO;
- A instrução trap (software) comuta de modo usuário para modo kernel e inicia o SO e quando termina, o controle é retornado ao usuário;
- Os dispositivos de E/S e CPU podem executar concorrentemente;
- Cada controlador tem um buffer local (área de armazenamento temporário que mantém dados durante E/S entre dispositivos com velocidade diferente).
- A CPU move dados de/para memória principal e para/de buffers locais;

Unidade Central de Processamento (Central Process Unit-CPU)

- Unidade de Controle (UC)
- Unidade Lógica e Aritmética (ULA)
- Clock (pulsos/ciclos)
- Registradores – armazena dados do estado do programa.
  - **Contador de instruções** – contém o endereço da próxima instrução que o processador deve buscar e executar.
  - **Apontador de pilha** – contém o endereço de memória do topo da pilha (LIFO - last in, first out), estrutura de dados que mantém informação sobre o programa que está sendo executado e foram interrompidos.
  - **Registrador de Status** – responsável por armazenar informações sobre a execução de instruções (overflow).

Pipelining

- Pipeline em 4 estágios (linha de montagem): A organização do processador em subsistemas funcionais resolve o problema de **ociosidade** devido a diferença de tempo de execução entre instruções;
- Executar instruções seguindo o mesmo modelo de linhas de produção;

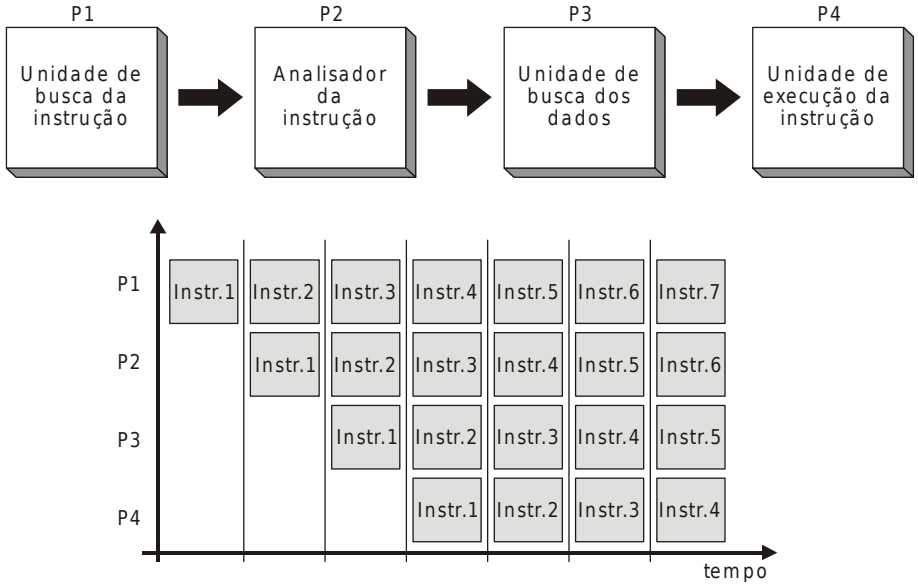


Figura 2: Etapas de um pipeline

Estrutura de Sistema de Computação

Interrupções

- Forma de melhorar a utilização de um processador;
- São geradas pelos programas, timer, E/S ou falhas;
- Uma tabela de ponteiros é usado para os serviços de interrupções:
  - *O vetor de interrupção* é indexado e contém os endereços de todas as rotinas de serviços.
  - Geralmente esse vetor fica localizado no início do espaço de memória.
- Geralmente, as interrupções entrantes são *desabilitadas* enquanto outra interrupção está sendo processada para prevenir uma *interrupção perdida*.

Funções Comuns de Interrupções

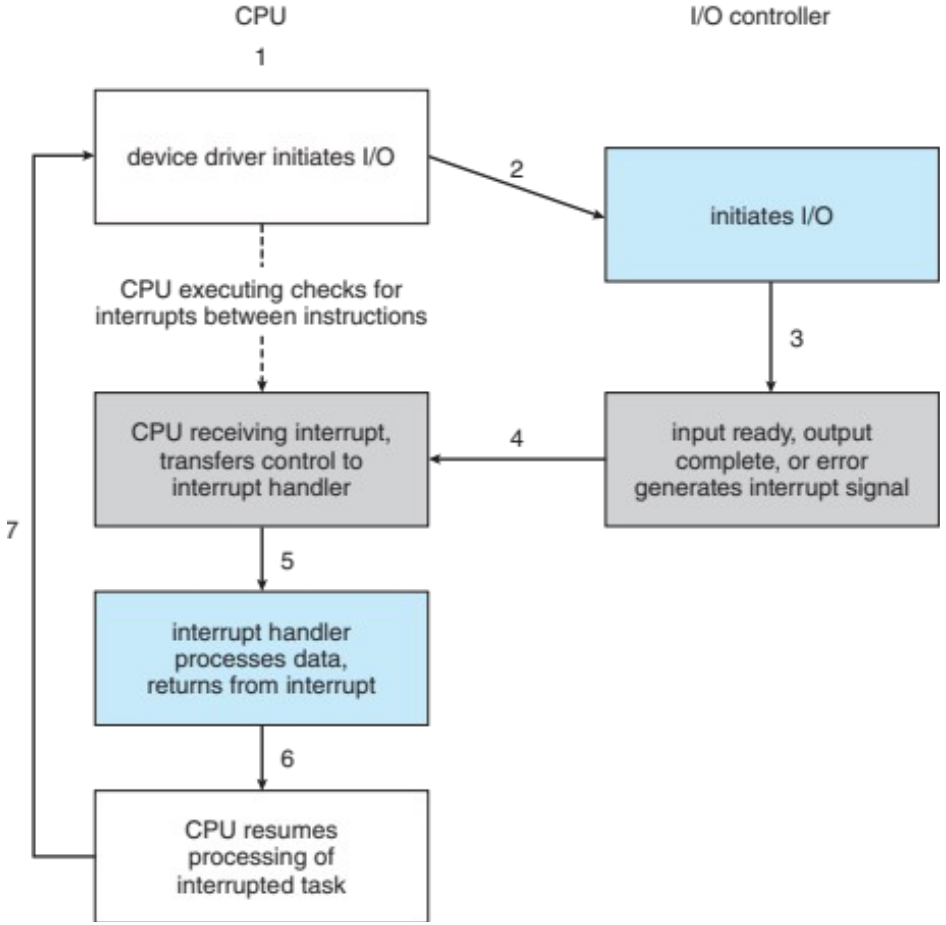
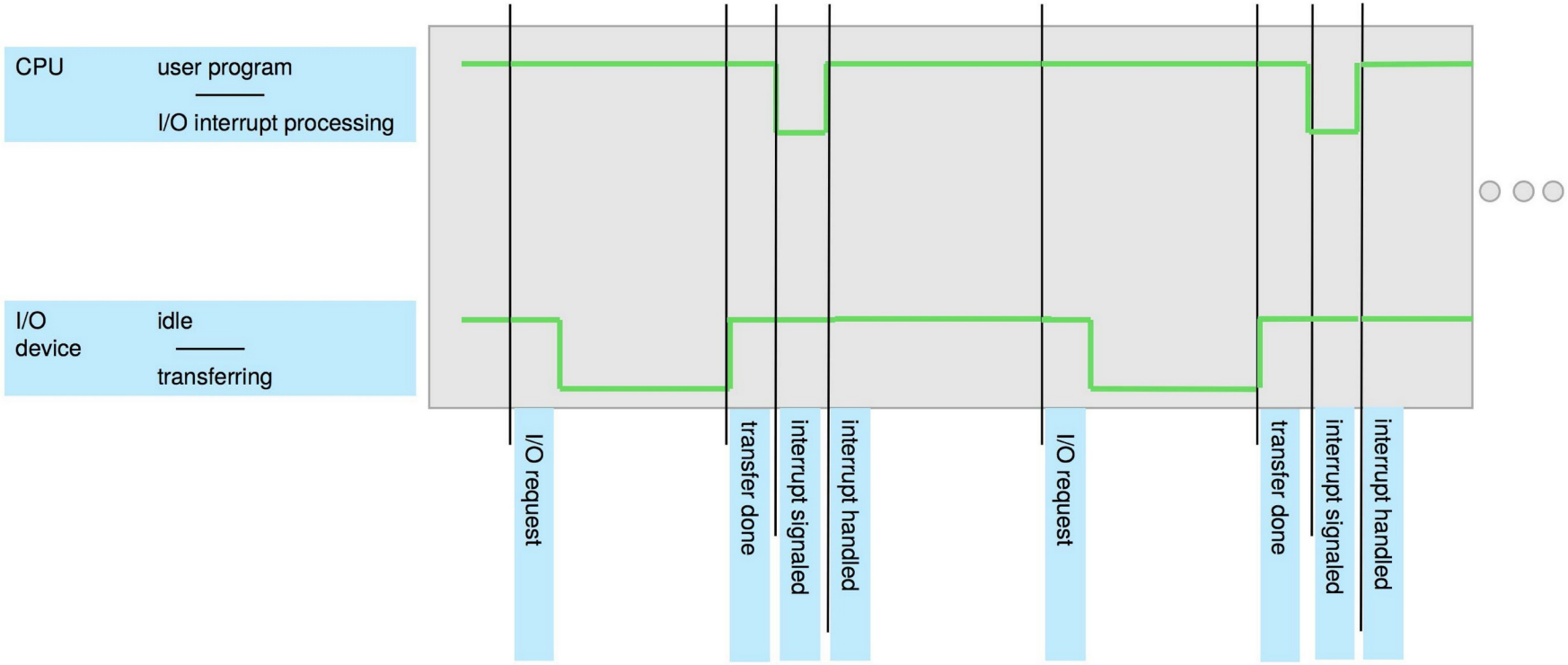


Figura 3: Etapas na execução de uma interrupção pelo programa

Timeline de Interrupções

Interrupções

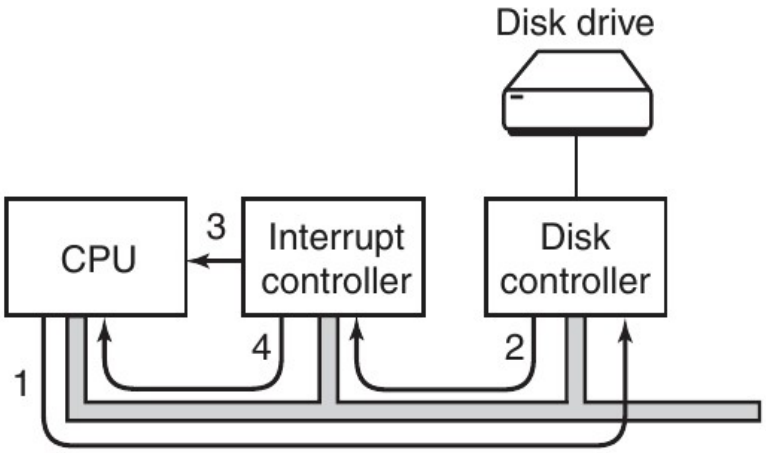


Interrupções

- O SO preserva o estado da CPU armazenando os registradores e o contador do programa;
- Se a interrupção é gerada por software é conhecida como *trap*;
- Um sistema operacional é um sistema movido a *interrupção*.
- Segmentos separados de código determinam quais ações serão tomadas para cada tipo de interrupção.

Estruturas de Sistemas de Computação

Exemplo de uma interrupção gerado para uso de disco.



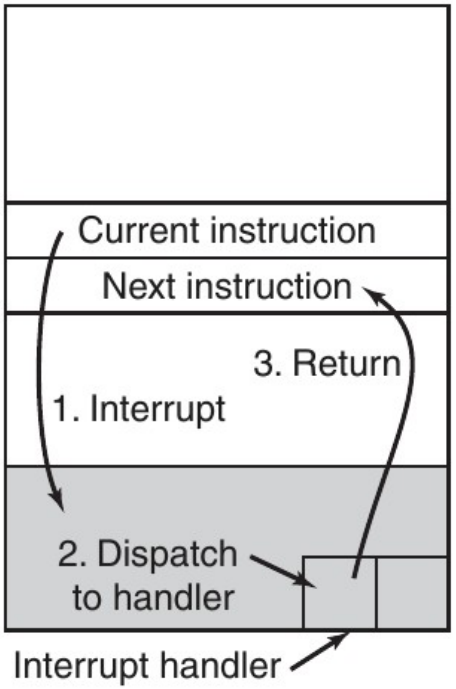


Figura 4: Etapas de uma interrupção de um dispositivo de E/S

Estrutura de Sistema de Computação

Estrutura DMA (Direct Memory Acess)

- Recurso de hardware que permite a realização de E/S programado a fim de manter a CPU ocupada única e exclusivamente com as atividades que requerem processamento de resultados.

Estrutura de Sistema de Computação

Estrutura DMA (Direct Memory Acess): O DMA realiza uma transferência de VÁRIOS blocos de dados de um dispositivo para a memória, sendo que a CPU só interrompida (interrupção) quando todos estes blocos estiverem no local desejado.

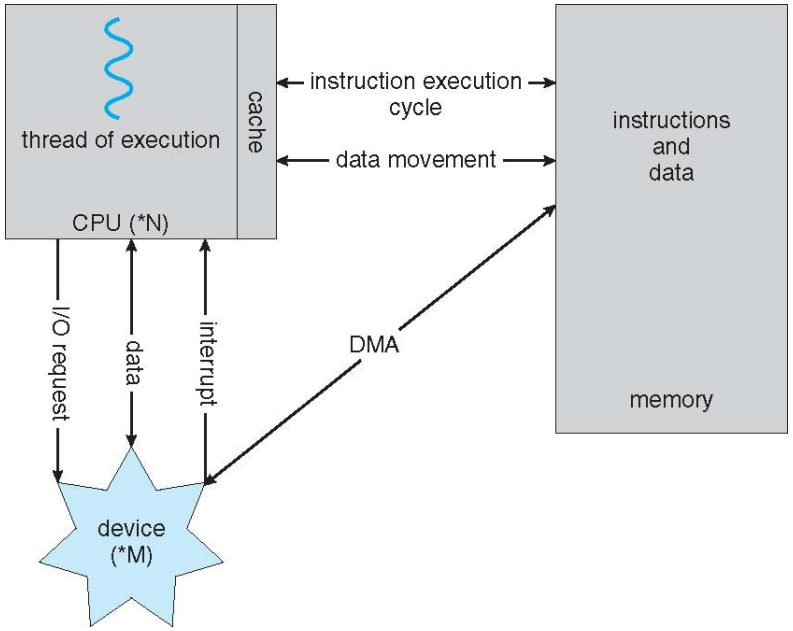


Figura 5: Etapas durante o uso do recurso DMA

Sumário

- Hierarquia de Armazenamento
- Proteção de Hardware
- Virtualização
- Arquitetura

Estruturas de Sistemas de Computação

Hierarquia de Armazenamento

- Sistemas de Armazenamento são organizados em hierarquia:
  - Velocidade
  - Custo
  - Volatilidade
- *Caching: copia* informação para um sistema mais rápido de armazenamento;
- A memória principal pode ser vista como um *cache* rápido para o armazenamento secundário (disco rígido).

Estrutura de Sistema de Computação

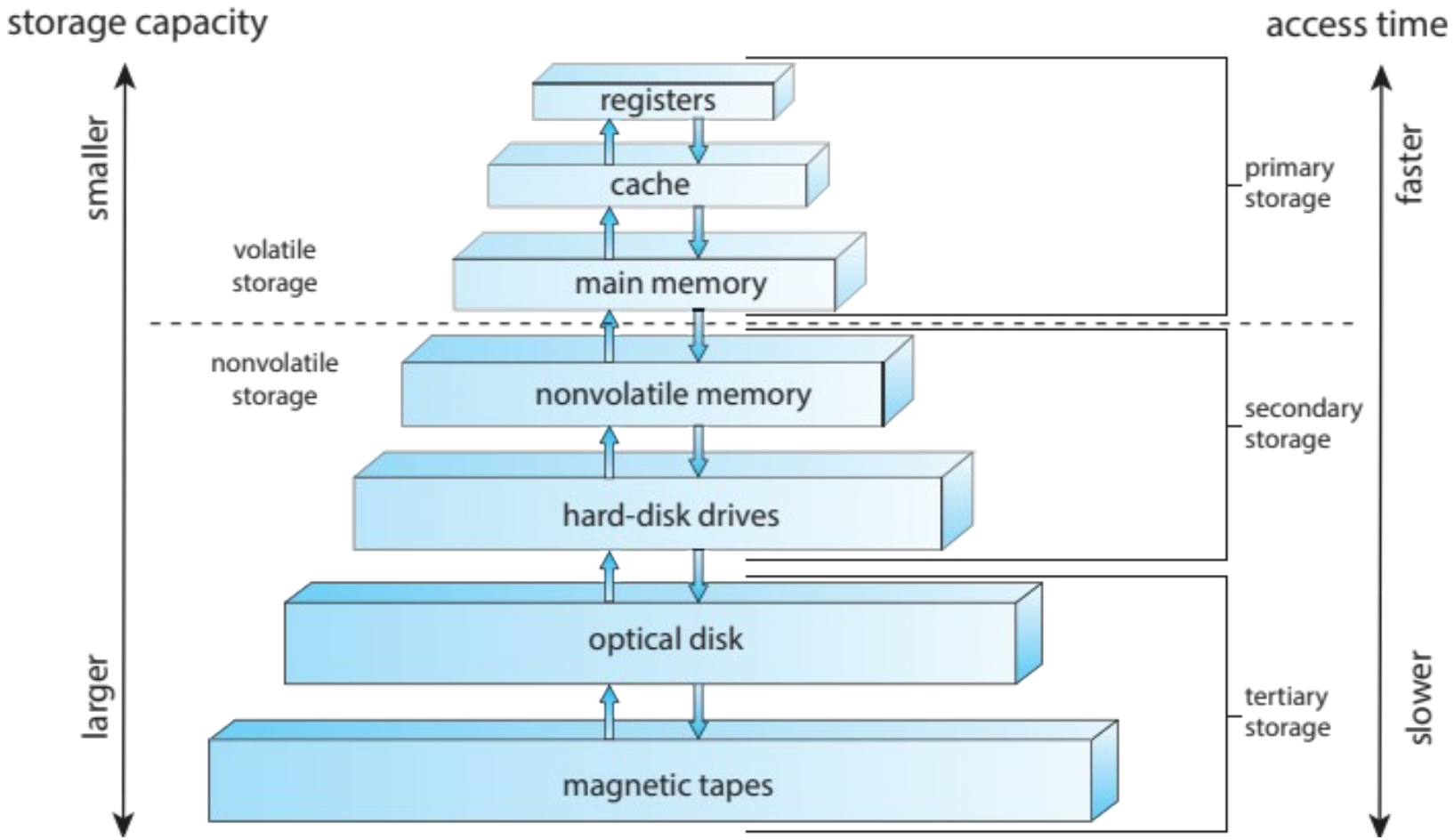


Figura 6: Hierarquia de tipos de memória em um sistema computacional

Estrutura de Sistema de Computação

Armazenamento

- $ /proc/meminfo
- $ free -m
- $ htop
- Fornece informações sobre processos, memória, páginas, blocos e CPU: $ vmstat
- $ gnome-system-monitor
- Arquivos abertos: $ lsof

Estrutura de Armazenamento Secundária

- Discos magnéticos: pratos de vidro ou de metal rígido revestidos de material magnético para gravação;
- A superfície do disco esta logicamente dividida em trilhas (*tracks*), as quais estão divididas em setores (*sectors*);
- O controlador do disco determina a interação lógica entre o dispositivo e o computador.

Estrutura de Armazenamento

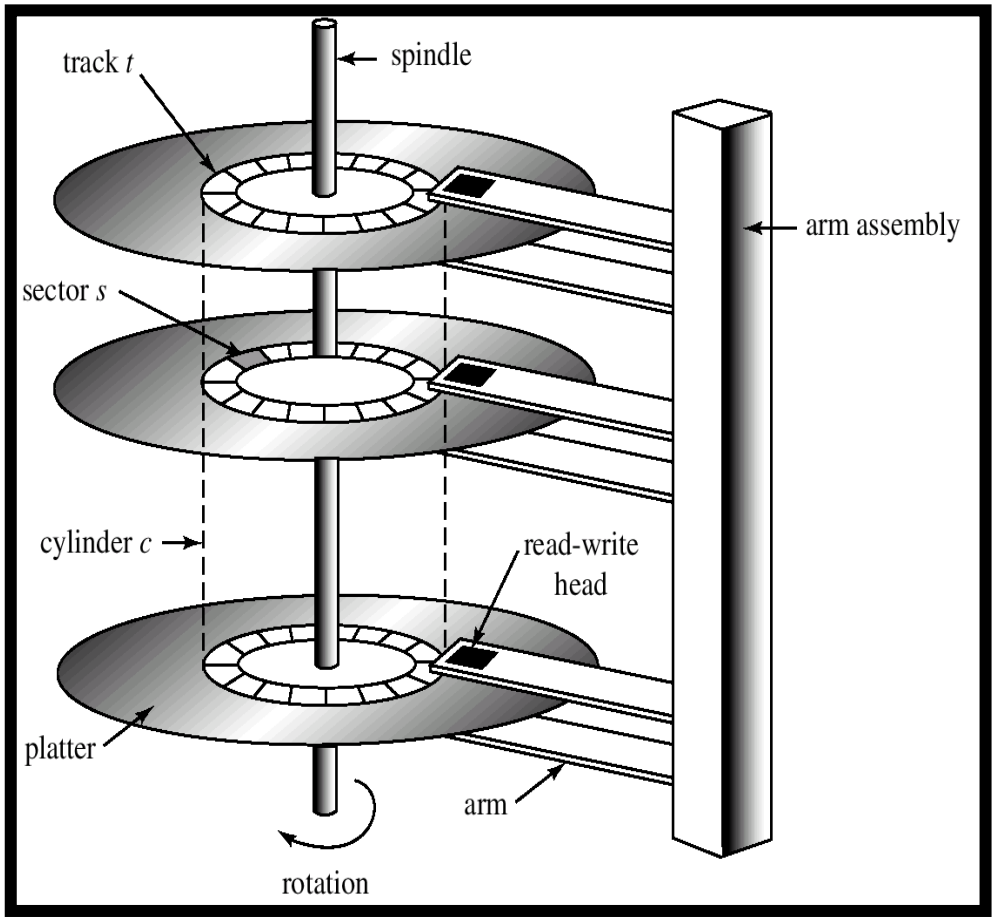


Figura 7: Estrutura de um disco rígido.

Estrutura de Armazenamento

- Solid-State Disks (SSD): é uma memória não volátil que é usada como um *hard drive*;

- Tem características semelhantes aos *hard disks*, mas são mais eficientes por não ter movimento de partes e não precisar de tempo de busca;

- Entretanto, são mais caros por megabyte do que os *hard disks*, têm menor capacidade, e pode ter períodos de vida mais curtos do que os discos rígidos.

Sumário

- Operações (Proteção de Hardware)
- Virtualização
- Arquitetura

Operações de Sistema Operacional

- Uma importante característica dos sistemas operacionais modernos é a habilidade de operação em multiprogramação;
- A multiprogramação aumenta uso de CPU e deve garantir que sempre de forma segura uma tarefa em execução.

Proteção:

- Operação em modo dual
- Proteção de E/S
- Proteção de Memória
- Proteção de CPU (timer)

Operações de Sistema Operacional

Operação em Modo Dual

- Compartilhamento de recursos do sistema obriga o SO a garantir que um programa *errado* não provoque execução errônea de outros programas;
- Provê suporte do hardware para diferenciar entre dois **modos de operação**:
  - *Modo usuário:* execução feita em nome de um usuário;
  - *Modo Monitor:* (também *modo supervisor* ou *modo sistema*) − execução feita em nome do SO.

Operações de Sistema Operacional

Modo Dual:

- ***Mode bit*** adicionado ao hardware do computador para indicar o modo corrrente: monitor (0) ou usuário (1).
- Quando uma interrupção ou erro ocorre o hardware troca para o modo monitor.

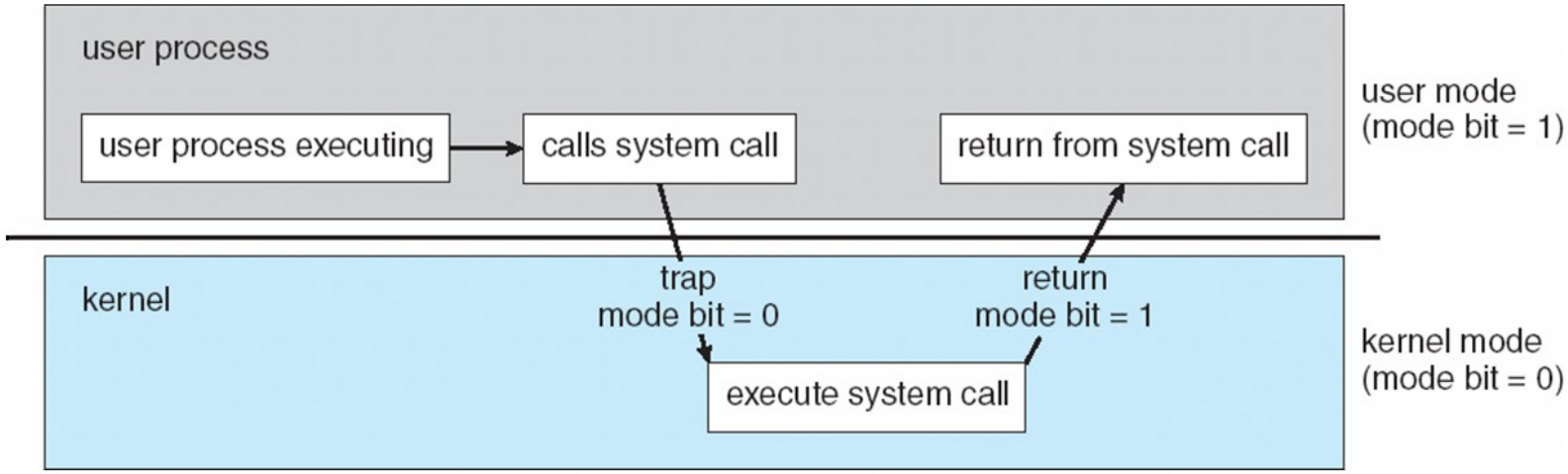


Figura 8: Mudança de modo de operação

Operações de Sistema Operacional

Proteção de E/S

- Todas as instruções de E/S são privilegiadas.
- Deve garantir que um programa de usuário nunca possa ganhar controle do computador no modo monitor (i.e., um programa de usuário que, como parte da sua execução, armazena um novo endereço no vetor de interrupção).

Proteção da Memória

- Deve prover proteção da memória no mínimo para o vetor de interrupção e as rotinas de serviço de interrupção.

Proteção da Memória (cont)

- Para proteção existe 2 registradores que determinam a faixa de endereços legais que um programa pode acessar:
  - **base register** contém o menor endereço de memória física legal.
  - **limit register** contém o tamanho da faixa.
- Memória fora da faixa definida é protegida.

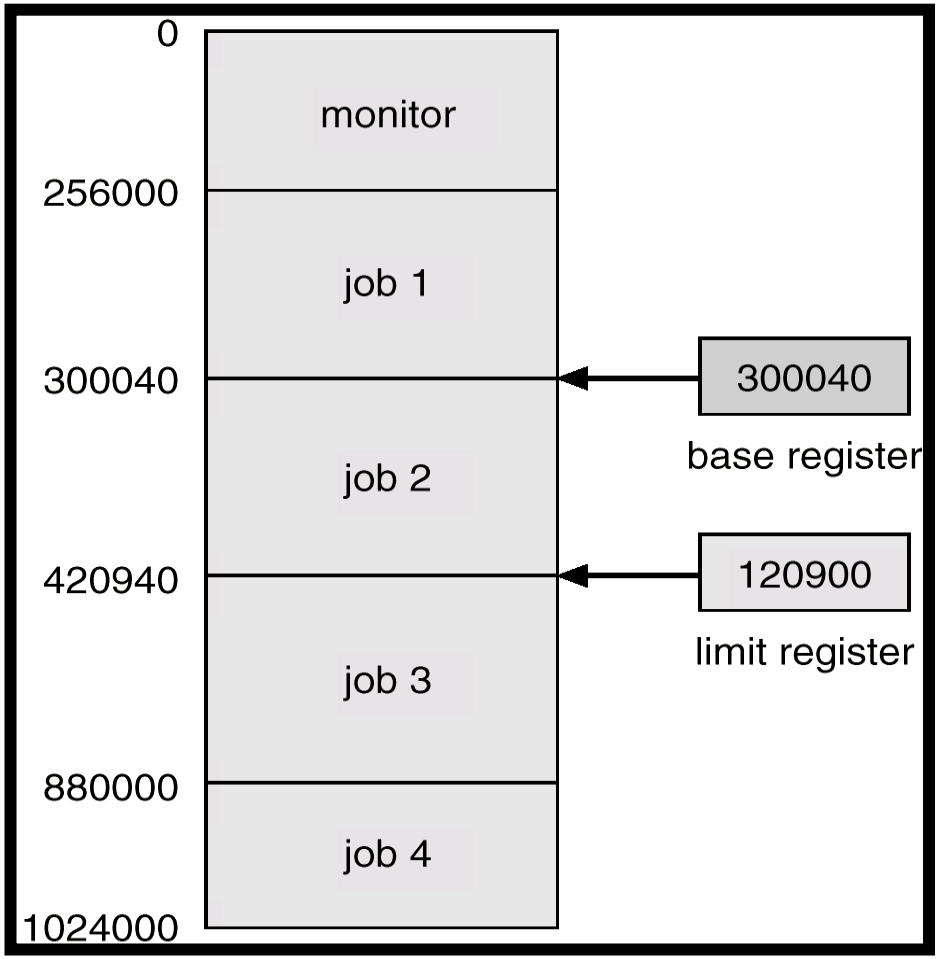


Figura 9: Endereço de memória física

Operações de Sistema Operacional

Proteção da CPU

- ***Timer:*** interrompe o computador após um período especificado de tempo para garantir que o SO mantém o controle (podendo ser um período fixo ou variável com mecanismo de decremento);
  - Timer é decrementado a cada *tick* do relógio.
  - Quando o timer chega a “0” uma interrupção ocorre.
- O Timer é comumente usado para implementar compartilhamento de tempo;
- Timer também utilizado para calcular a hora corrente.
  - *(2 timers: um para CPU (SO) e outro para Hora).*
- Carregar o *timer* é uma instrução privilegiada.

Sumário

- Virtualização
- Arquitetura

Virtualização

- Permite sistemas operacionais executar aplicações dentro de outros sistemas operacionais;
- **Emulação** usado quando o tipo de CPU difere do tipo alvo (exemplo: PowerPC para o Intel x86)
  - Geralmente métodos mais lentos;
  - Quando a linguagem de computador não compila o código nativo – **Interpretação.**
- **Virtualização** – SO de forma nativa compilado para a CPU, executando **guest SO** também de forma nativa compilado
  - Considere o VMware executando o convidado Linux, cada aplicação executando, todas sobre o SO WinXP anfitrião;
  - **VMM** (virtual machine Manager) fornece serviços de virtualização;

Máquinas Virtuais

- A ideia é separar o hardware em vários ambientes de execução diferentes;
- O SO cria a ilusão de múltiplos processos: cada um executando na seu próprio processador com sua própria memória;
- Os recursos do PC são compartilhados para criar as máquinas virtuais.
  - Escalonamento da CPU pode criar a aparência de que os usuários têm seus próprios processadores;
  - Sistema de Arquivos e *Spooling* provêm impressoras virtuais, etc.

Máquinas Virtuais

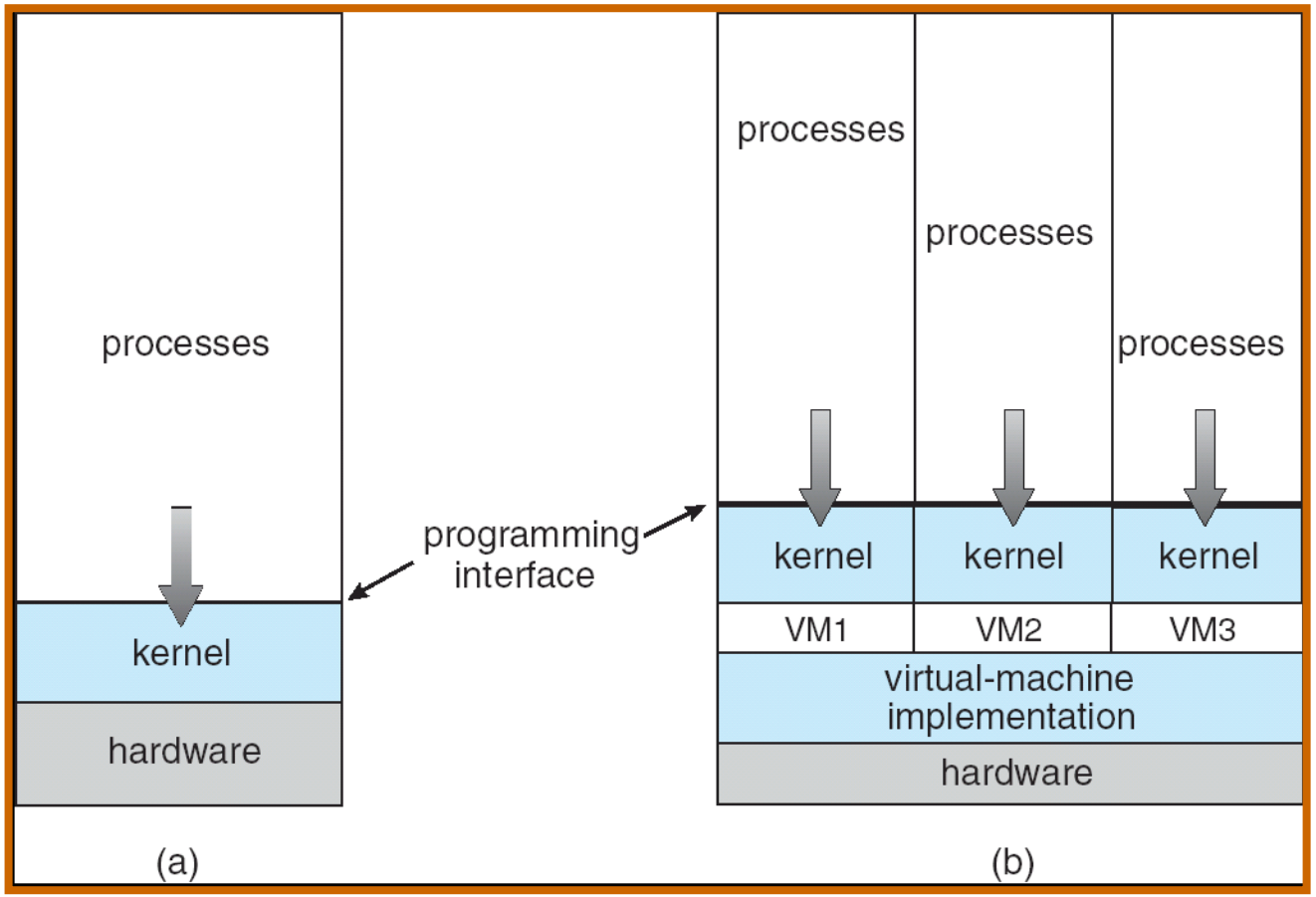


Figura 10: (a) máquina não virtual e (b) máquina virtual.

Máquinas Virtuais

Porque usar máquina virtual?

- Em pesquisas e desenvolvimento:
  - Testes de SO em MV com várias configurações;
  - Tarefa de mudar o SO: mais rápido testar em MV;
  - MV fornece proteção completa do sistema pesquisado;
  - Cada máquina é isolada de outras e dificilmente compartilha recursos.
- Exemplos:
  - Virtual Box, Vmware, Java Virtual Machine

Máquinas Virtuais

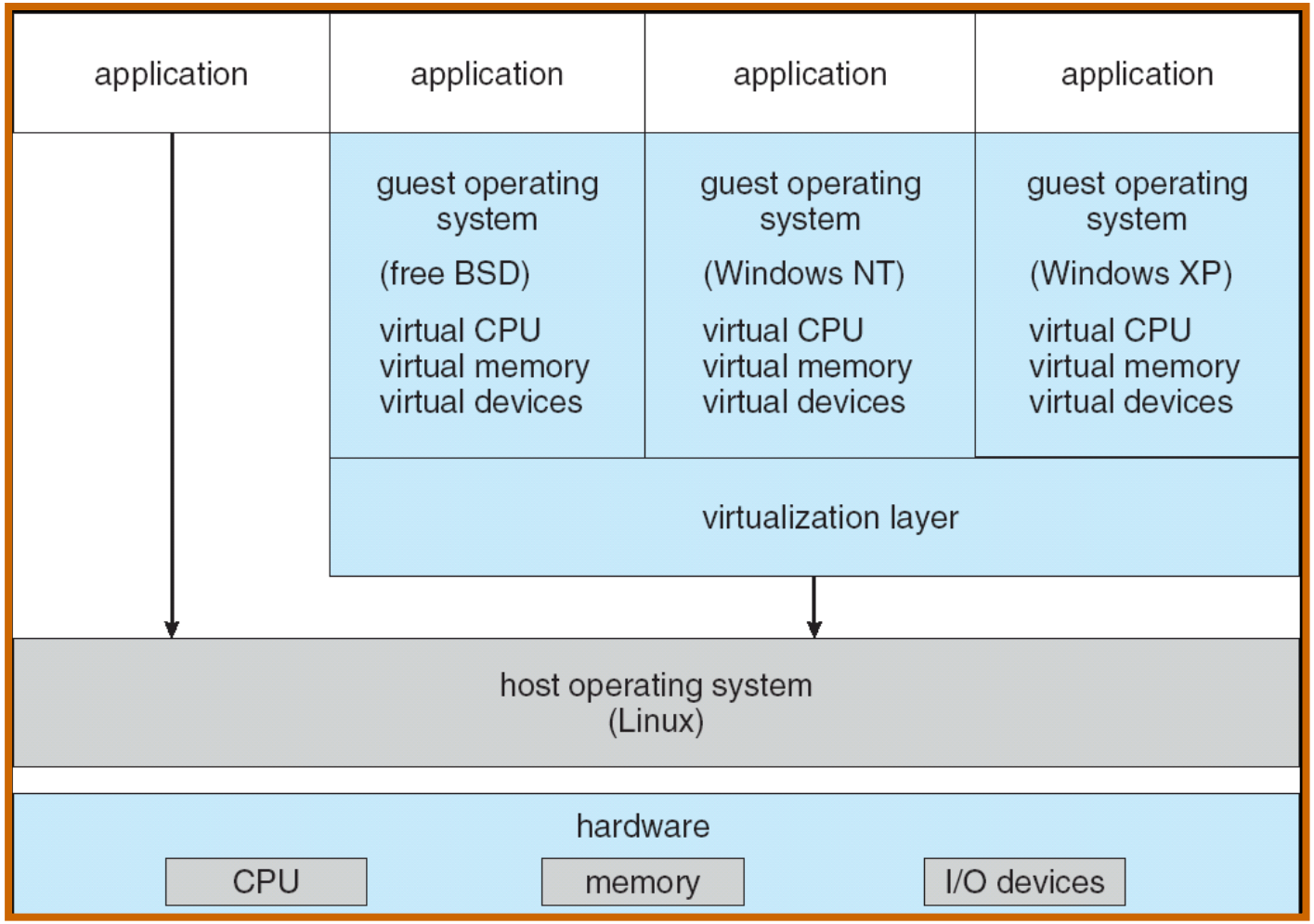


Figura 11: Arquitetura do VMware.

Máquinas Virtuais

Java

- Especificação para computador abstrato;
- Consiste em um carregador de classes e um interpretador que executa os códigos de bytes independente da arquitetura.

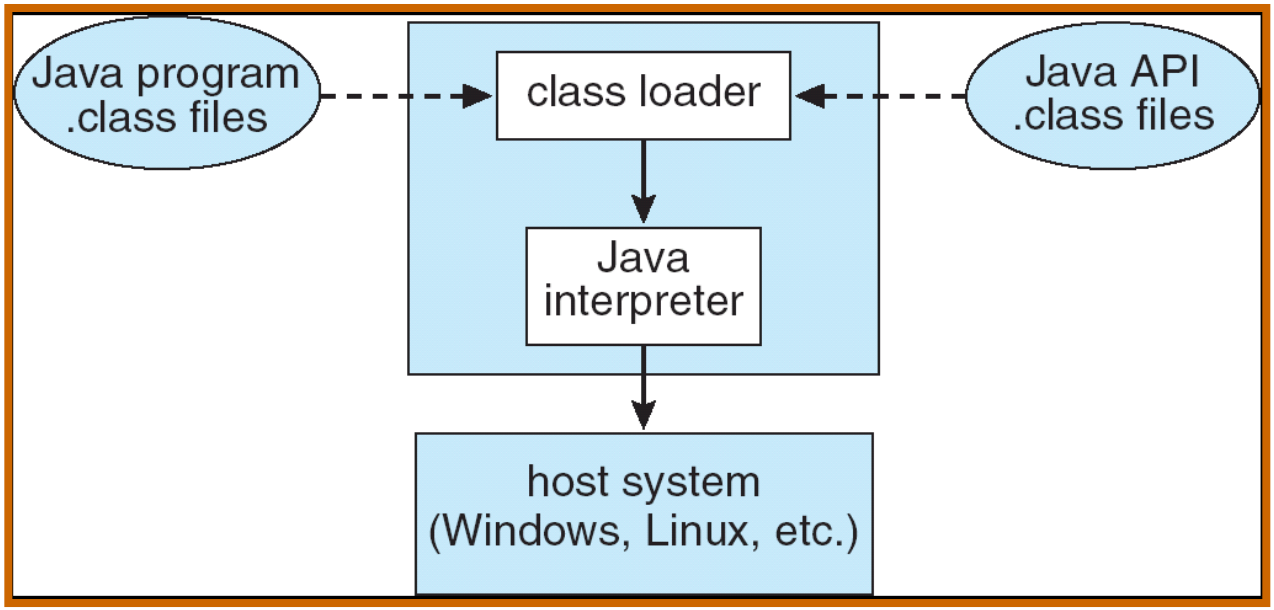


Figura 12: A máquina virtual Java.

Ambiente de Desenvolvimento em Java

- Ambiente em tempo de compilação;
- Ambiente em tempo de execução.

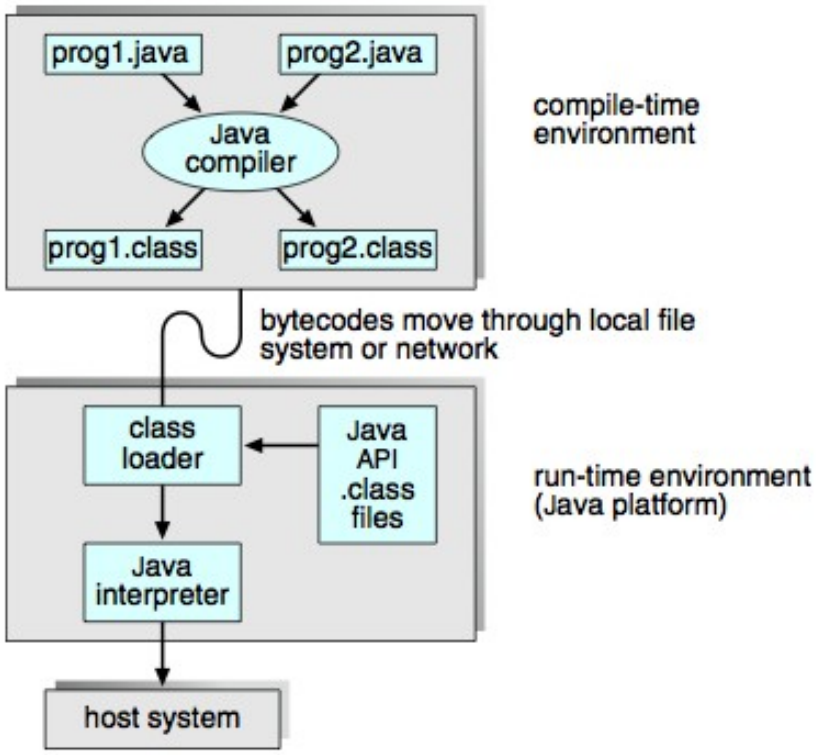


Figura 13: Etapa em um ambiente Java

Sumário

- Arquitetura

Arquitetura de Sistema Computacional

- Muitos sistemas usam um simples processador
  - Muitos sistemas tem processador de proposito especial
- **Multiprocessadores** vêm crescendo em uso e importância
  - Também denominado como sistemas paralelos ou sistemas fortemente acoplados
  - Vantagens:
    1. **Aumento de throughput**
    2. **Economia**
    3. **Maior confiabilidade** – em relação a degradação ou tolerância a falha
  - Tipos:
    1. **Assimétrico** – cada processador é alocado para uma tarefa específica.
    2. **Simétrico** – cada processador pode participar da execução de todas as tarefas

Arquitetura: Simétrica

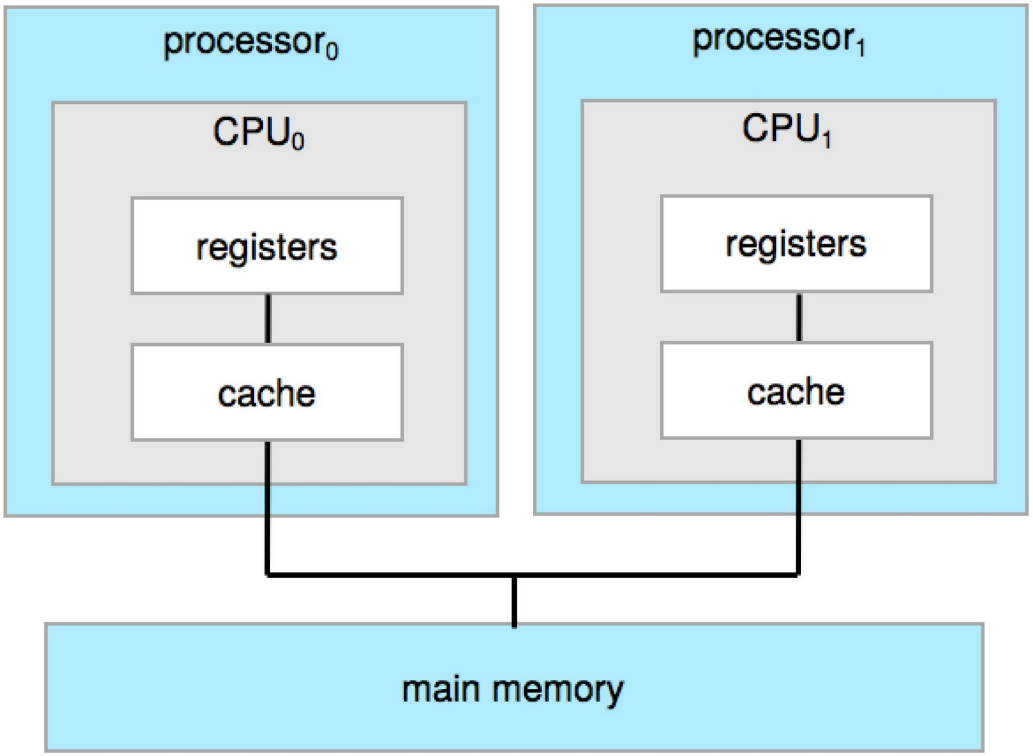


Figura 14: Arquitetura Simétrica

Arquitetura: Dual Core

- Múltiplos circuitos e múltiplos núcleos
- Sistema contém todos os circuitos
  - Chassi contém múltiplos sistemas separados

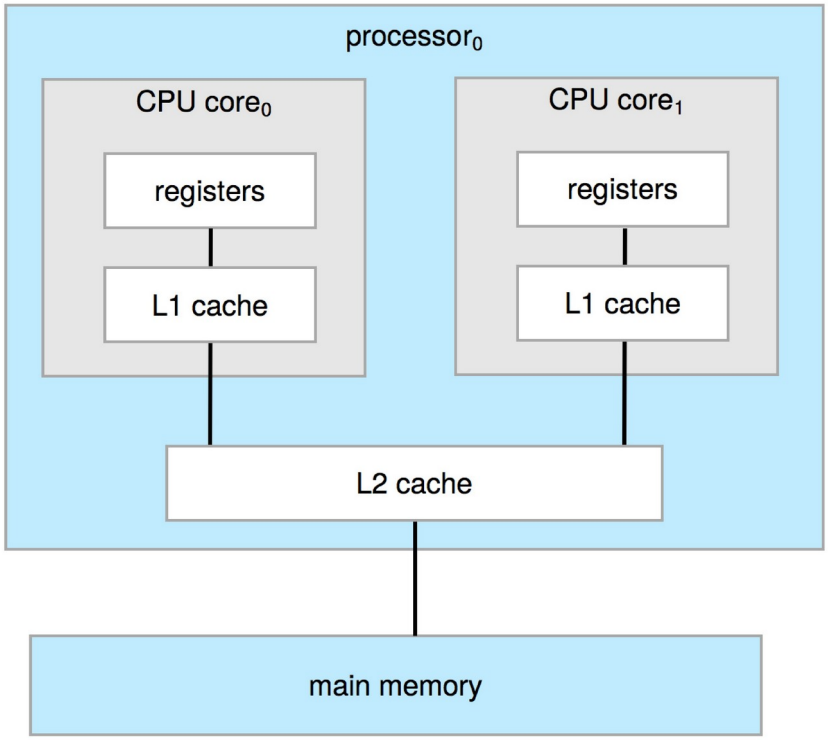


Figura 15: Arquitetura Dual Core

Sistema de acesso ao memória não uniforme

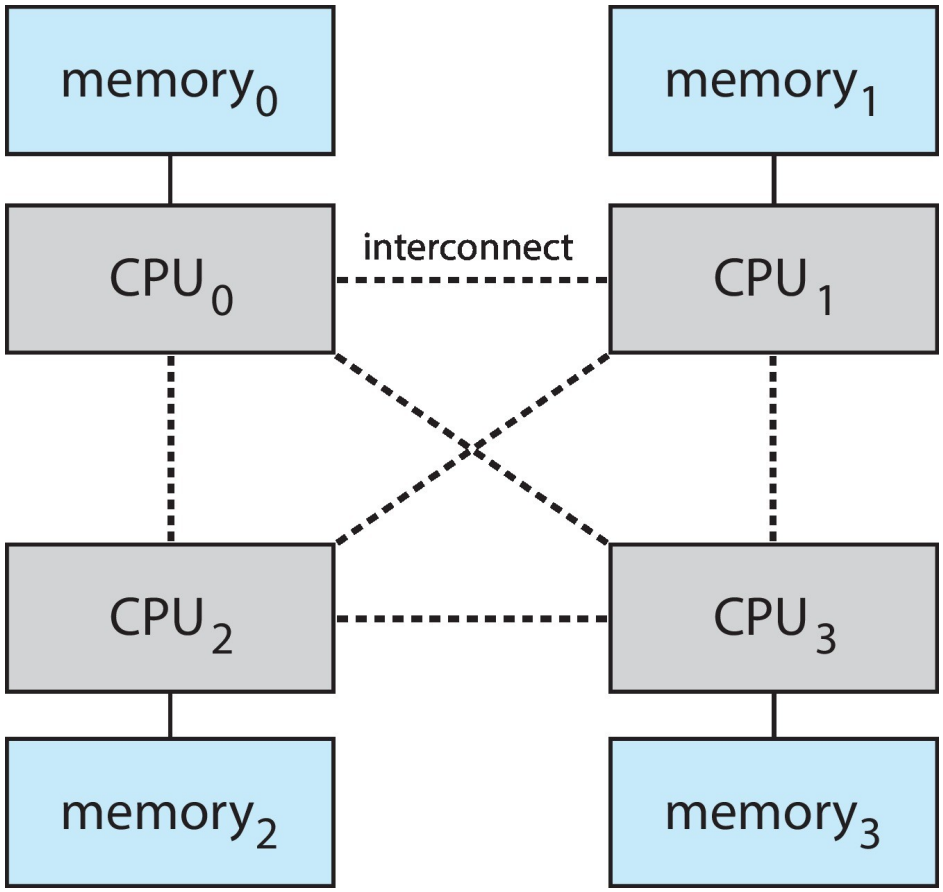


Figura 16: Acesso a memória não uniforme

Arquitetura: Cluster

- Semelhante ao sistema multiprocessadores, mas com múltiplos sistemas trabalhando juntos
  - Normalmente compartilha armazenamento via uma **storage-area network (SAN)**
  - Fornece um serviço de alta disponibilidade
    - **Assimétrico** tem uma máquina no modo hot-standby
    - **Simétrico** tem múltiplos nós executando aplicações
  - Alguns cluster são **high-performance computing (HPC)**
    - Aplicações deve ser escritas usando a paralelização
  - Alguns tem **distributed lock manager (DLM)** para evitar conflito em operações

Arquitetura: Cluster

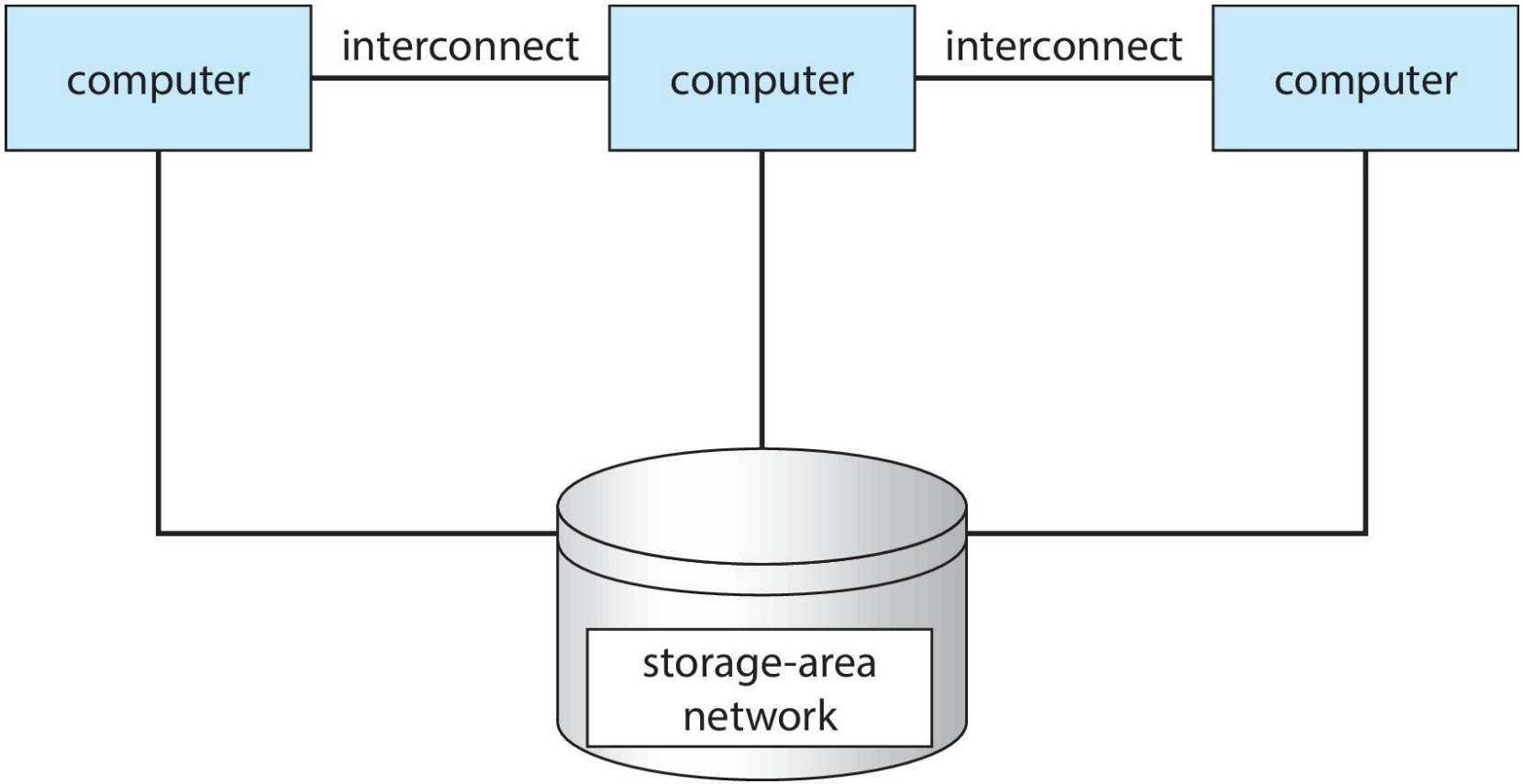


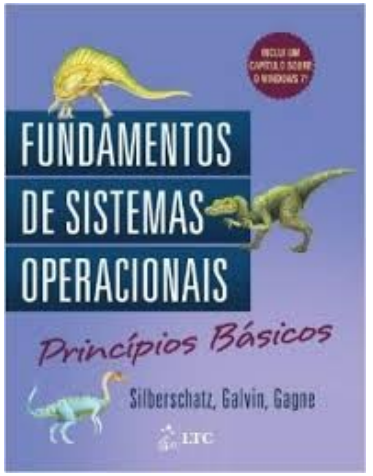
Figura 17: Modelo Cluster

Sumário

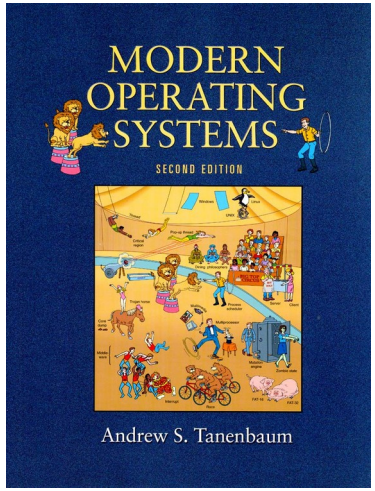
- Estrutura de Sistema Computacional
  - Sistema de Boot
  - CPU
  - Interrupções
  - DMA
- Hierarquia de Armazenamento
- Proteção de Hardware
- Virtualização
- Arquitetura
- Leituras Sugeridas

Leituras Sugeridas

- SILBERSCHATZ, A. & GALVIN, P. B. & GAGNE, G. Fundamentos de Sistemas Operacionais. 8ª Edição, Ed. LTC.



- Andrew S. Tanembaum. **Sistemas Operacionais. Modernos**. 2ª Ed. Editora Pearson, 2003.



Lista de Exercícios

- Lista 01:
- Exercícios até o número 39.