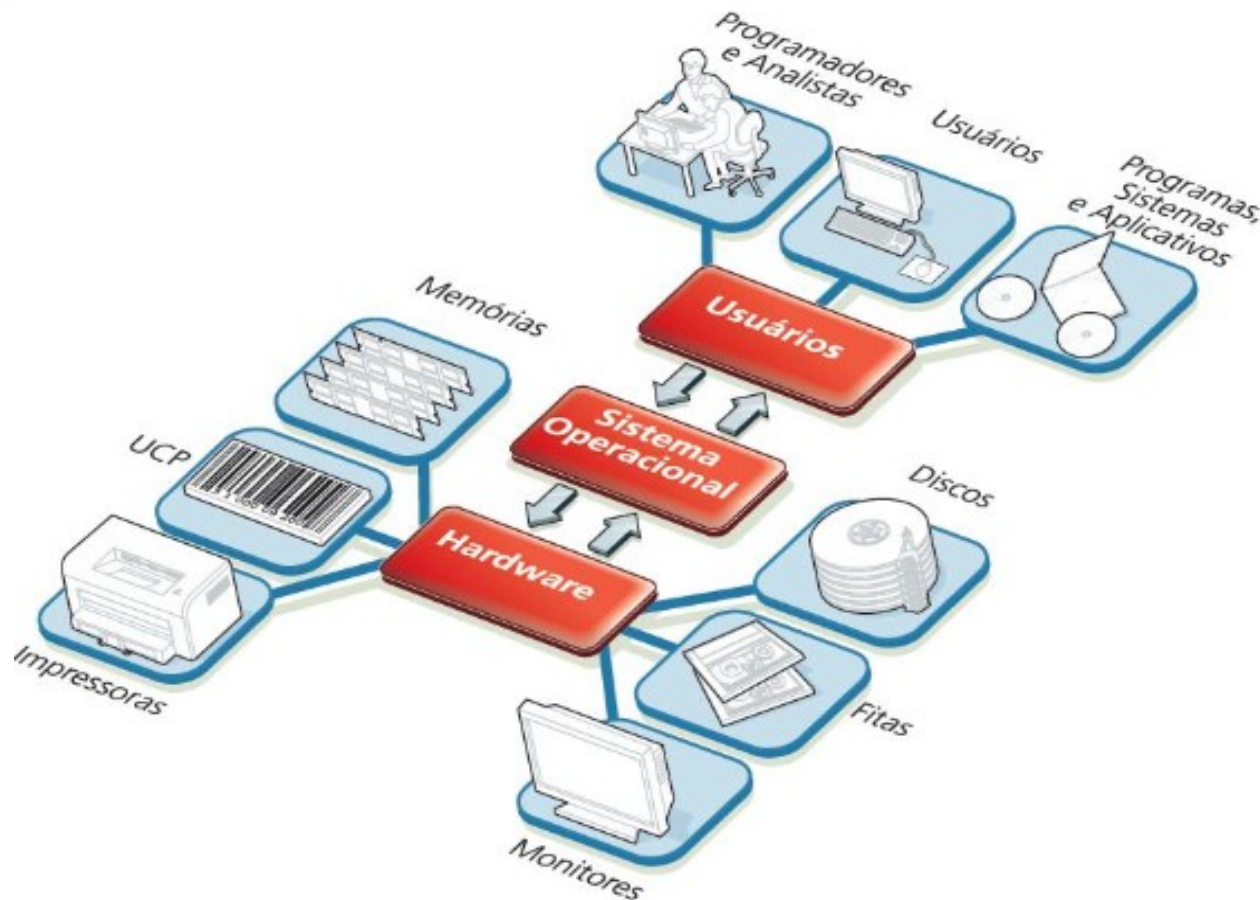


INSTITUTO FEDERAL  
PIAUÍ  
Campus Parnaíba

# Estrutura dos SO's

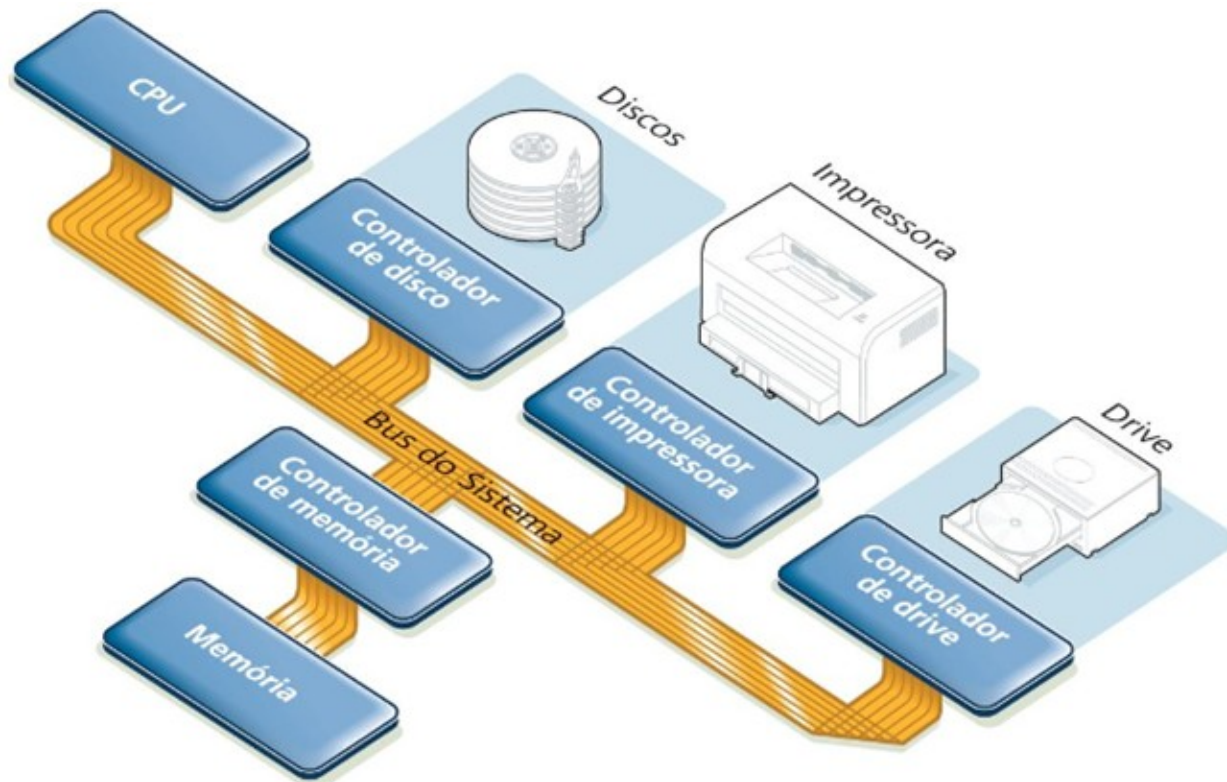
Prof. Msc Denival A. dos Santos

# Estrutura do computador

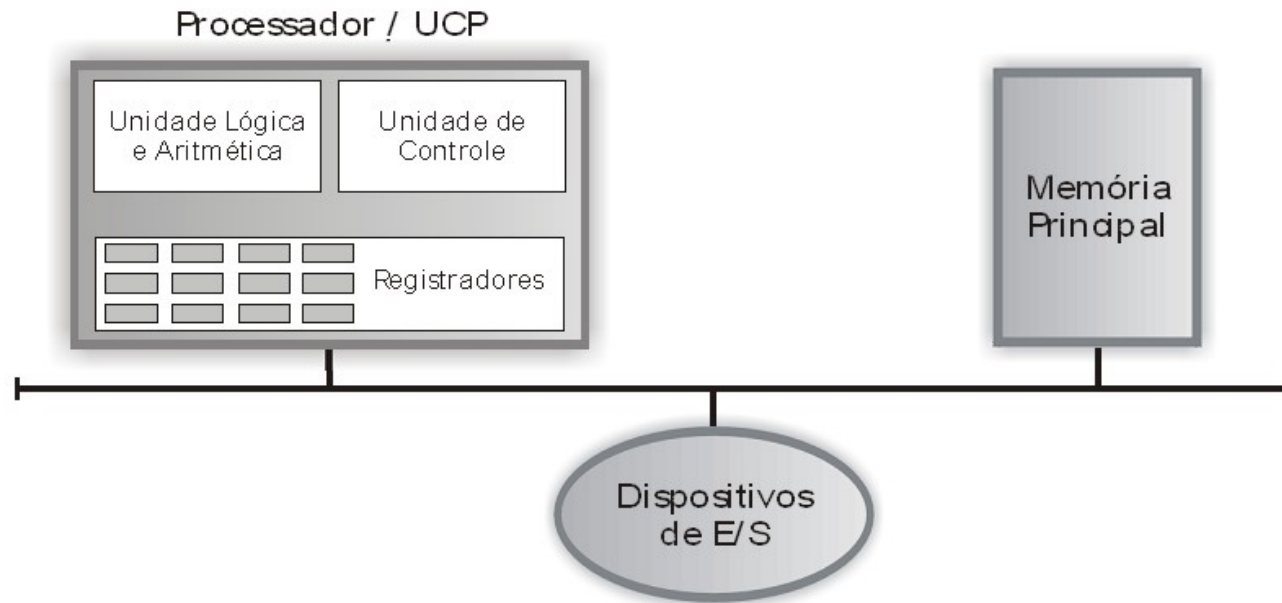


- Os principais elementos constituintes do computador (processador, memória e controladores de periféricos) estão interligados por um ou mais barramentos (para a transferência de dados, endereços e sinais de controle).

# Estrutura do computador



# Estrutura do computador



- Um sistema computacional geralmente é formado pela composição de três subsistemas básicos (também conhecidos como unidades funcionais): processadores (UCP), memória principal (RAM) e dispositivos de entrada e saída (E/S). Estes itens compõem qualquer tipo de computador digital existente independente de fabricante, marca ou modelo.

# Estrutura do computador



## ■ Processador

- O processador ou Unidade Central de Processamento (UCP) tem como função controlar e executar instruções presentes na memória principal (RAM) através de operações aritméticas (como somar, subtrair, multiplicar e dividir) e operações lógicas (comparações, movimentação de dados, etc.).
- Um processador é composto por alguns elementos básicos como a Unidade de Controle (UC), a Unidade Lógica e Aritmética (ULA) e os registradores.
  - A UC gerencia as atividades de todos os dispositivos do computador, como por exemplo, a gravação de dados em discos, a busca de instrução em memória, entre outros.
  - Já a ULA realiza as operações lógicas e aritméticas, conforme citado nos parágrafos anteriores. Os registradores por sua vez têm a função de guardar dados temporariamente, servindo como uma memória de alta velocidade interna do processador, porém com uma capacidade de armazenamento reduzida e um custo maior que a memória principal.

# Estrutura do computador

## ▪ Memória Principal

- Um computador em seu interior possui vários tipos de memórias. A memória principal do computador é conhecida como **memória RAM** (*Random Access Memory*), ou memória de acesso aleatório. Ela é chamada de memória principal, pois é utilizada para manter os programas em execução, tanto dos usuários como do sistema operacional.
- A memória é composta por unidades de acesso denominadas de células. Cada uma das células possui um determinado número de *bits*.
- A memória RAM é volátil, ou seja, todo o conteúdo se apaga quando o computador é desligado ou reiniciado.
- A quantidade de memória RAM disponível em um computador, exerce um grande efeito sobre o desempenho do mesmo.
- Os *chips* de memória RAM são vendidos na forma de pentes de memória, com diferentes velocidades de funcionamento (atualmente na casa dos Gigahertz), tamanhos (01, 02, 04 GB, etc.) e tipos (DDR, DDR2, DDR3, etc.).

# Estrutura do computador

## ▪ Memória Cache

- A memória *cache* pode ser caracterizada como uma memória de alta velocidade, volátil e na grande maioria das vezes com pequena capacidade de armazenamento.
- Sua principal função é diminuir a disparidade existente entre a velocidade na qual o processador executa instruções, bem como a velocidade com que os dados são lidos e gravados em memória principal.
- O tempo de acesso a um dado presente na memória *cache* é muito menor que se o mesmo estivesse em memória principal.
- A maioria dos processadores atuais apresentam um esquema de memória *cache* composto por múltiplos níveis.
- Nesse esquema, o nível mais alto de *cache* é chamado de **L1** (*Level 1*), com baixa capacidade de armazenamento e com alta velocidade de acesso. O nível **L2** (*Level 2*) possui maior capacidade de armazenamento, porém velocidade de acesso inferior a L1. **L3** (*Level 3*) trabalha em sincronia com L1 e L2 para melhorar seu desempenho.

# Estrutura do computador

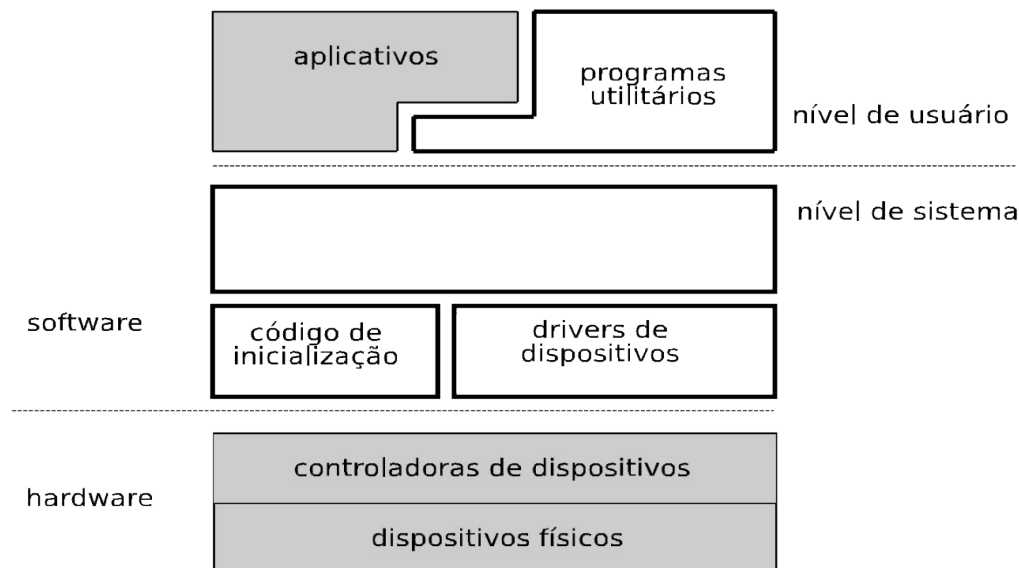
## ▪ Memória Secundária

- A memória secundária é utilizada para guardar dados de forma permanente no computador. Este tipo de armazenamento não necessita de alimentação, diferentemente da memória principal que necessita estar energizada para manter suas informações.
- O acesso à leitura e gravação de dados na memória secundária é mais lento se comparado à memória principal, entretanto o custo é baixo e a capacidade de armazenamento é bastante superior.
- Quanto aos tempos de acesso, a memória secundária funciona na ordem de milissegundos ( $10^{-3}$ ), enquanto a memória principal trabalha na ordem de nanossegundos ( $10^{-9}$ ).
- Podemos citar como exemplo de memória secundária os seguintes dispositivos: Discos magnéticos (discos rígidos - HD), Discos ópticos (CD, DVD, Blu-Ray), Memória *flash* (*pen drives*).



# Estrutura do Sistema Operacional

- Um sistema operacional não é um bloco único e fechado de software executando sobre o hardware. Na verdade, ele é composto de diversos componentes que têm objetivos e funcionalidades complementares.
- Os procedimentos do sistema são executados concorrentemente sem uma ordem específica ou predefinida, com base em eventos dissociados no tempo (eventos assíncronos).
- Alguns dos componentes mais relevantes de um sistema operacional típico são: núcleo, drivers, código de inicialização e programas utilitários.



# Operações do sistema de computação

- Os itens básicos contemplam a estrutura dos sistemas de computação são:
  - Inicialização do sistema
  - Entrada e saída (*input/output* - I/O)
  - Armazenamento
- Processadores e controladores de dispositivos podem operar concorrentemente, competindo por ciclos de memória. Para que um computador possa entrar em funcionamento (ligar) ou reiniciar é preciso basicamente de:
  - Um programa *bootstrap* (responsável por inicializar todos os aspectos do sistema, como registradores, controladores, entre outros. Este programa é armazenado na memória ROM do tipo *firmware*).
  - Ocorrência de eventos (interrupção proveniente tanto do *hardware* como do *software*).
- O sistema operacional espera que interrupções ocorram para a execução de processos, dispositivos de I/O para servir e a usuários a quem responder.

# Operações do sistema de computação

## ▪ Interrupções

- Uma **interrupção** é sempre causada por um evento externo ao programa que está em execução, ou seja, independe da instrução que está sendo executada.
- Um exemplo de interrupção é quando um dispositivo de *hardware* avisa ao processador que alguma operação de E/S está completa.
- As interrupções provêm de eventos assíncronos, ou seja, não estão relacionadas à instrução do programa corrente. Isto permite a ocorrência de múltiplas interrupções simultâneas.
- Uma interrupção é um evento tratável pelo sistema operacional.

# Operações do sistema de computação

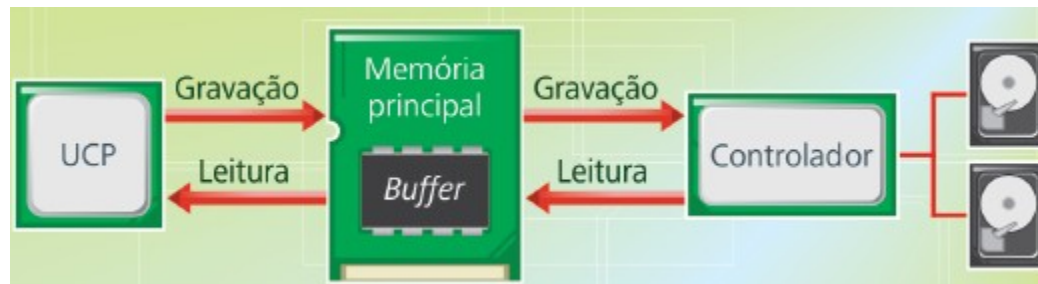
## ▪ Exceções

- Uma **exceção** é bastante semelhante a uma interrupção, sendo o que diferencia uma da outra é o motivo pelo qual o evento é gerado.
- Uma exceção pode ocorrer através de uma instrução do próprio *software*, como por exemplo, a divisão de um número por zero, entre outros.
- A exceção diferentemente da interrupção é gerada por um evento síncrono (resultado direto da execução do programa corrente).
- Na grande maioria das vezes, uma exceção provoca um erro fatal no sistema, causando o término anormal do programa.
- Da mesma forma que na interrupção, sempre que uma exceção é gerada o programa em execução é interrompido e o controle é desviado para uma rotina de tratamento de exceção

# Operações do sistema de computação

## ▪ *Buffering*

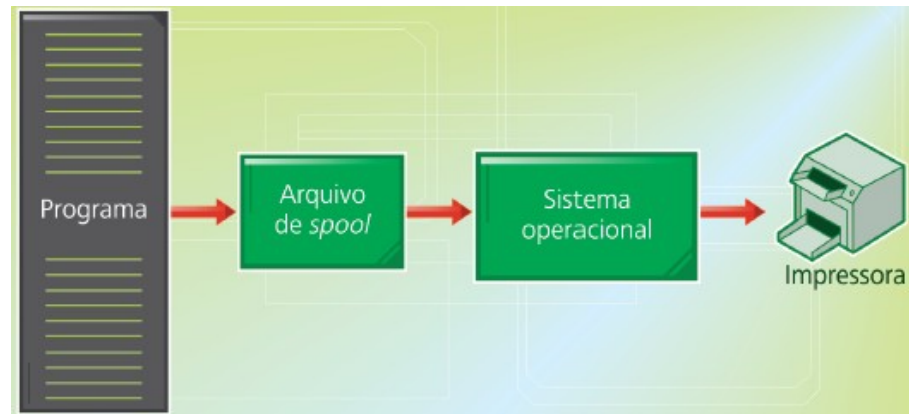
- A técnica denominada de *buffering* consiste em utilizar uma área de memória principal, chamada *buffer*, criada e mantida pelo sistema operacional. Possui a finalidade de auxiliar a transferência de dados entre dispositivos de E/S e a memória. O *buffer* permite minimizar a disparidade de velocidade entre o processador e os dispositivos de E/S e, tem como objetivo principal, manter tanto os dispositivos de E/S como o processador, ocupados a maior parte do tempo.
- O registro é a unidade de transferência do mecanismo de *buffering*. O *buffer* deve comportar o armazenamento de diversos registros, de forma que o processador tenha à sua disposição dados suficientes para processar sem ter que interromper o programa a cada leitura/gravação no dispositivo de E/S.



# Operações do sistema de computação

## ▪ *Spooling*

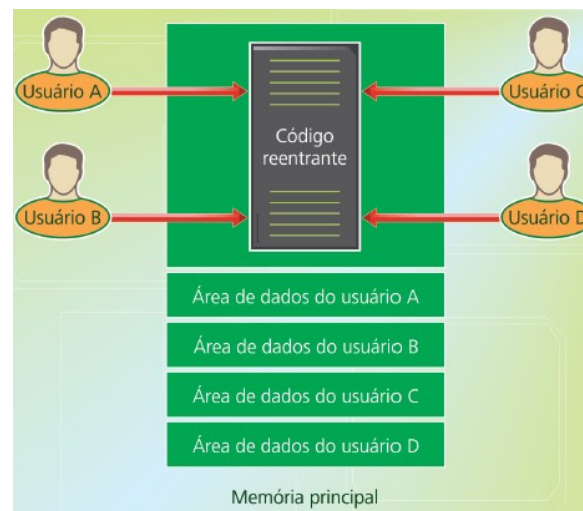
- A técnica de *spooling* foi criada inicialmente para auxiliar a submissão de processos ao sistema, sendo os processos gravados em fita para posterior leitura e execução. Com o aparecimento dos terminais para acesso ao sistema, esta técnica teve sua função adaptada para armazenar o resultado da impressão dos programas em execução.
- Isto é possível através da criação e manutenção, pelo sistema operacional de uma grande área em disco, com a finalidade de simular uma impressora. Assim, os usuários e seus programas imprimem, na verdade, para este arquivo em disco, liberando a associação dos dispositivos de impressão diretamente aos programas que estão executando.



# Operações do sistema de computação

## ▪ Reentrância

- Em sistemas multiprogramáveis, é normal que vários usuários utilizem os mesmos aplicativos simultaneamente, como editores de texto, compiladores, entre outros utilitários.
- A **reentrância** é a capacidade de um código executável (código reentrante) ser compartilhado por vários usuários, exigindo apenas uma cópia do programa em memória. Esta técnica permite que cada usuário esteja executando um trecho diferente do código reentrante, manipulando dados próprios, exclusivos de cada usuário. Assim, a reentrância tem como objetivo geral promover o uso mais eficiente da memória e um desempenho maior do sistema.



# Componentes de um Sistema Operacional

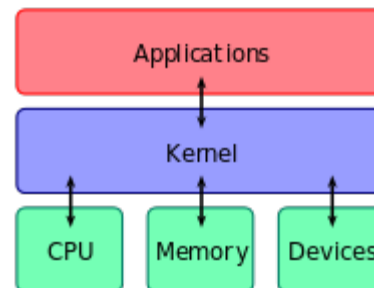
- Kernel (núcleo)
- Gerência de Processos
- Gerência de memória
- Gerência de Entrada e Saída
- Gerência de Arquivos
- Controle de usuários
- Gerência de janelas e GUI (interface gráfica)
- Gerência de redes
- Gerência de erros
- Contabilidade de uso de recursos
- Gerencia de segurança



# Componentes de um Sistema Operacional

## ▪ Núcleo (Kernel)

- É um conjunto de instruções (programa) do sistema operacional que fornece serviços aos usuários, às suas aplicações e também ao próprio sistema.
- Também chamado de Kernel é o coração do sistema operacional, responsável pela gerência dos recursos do hardware usados pelas aplicações.
- As aplicações são utilizadas pelos usuários de maneira transparente, escondendo todos os detalhes da interação com o sistema.
- Os usuários podem se comunicar com o kernel de 3 formas:
  - Rotinas de sistema: realizadas pelas aplicações.
  - Utilitários: compiladores, editores de texto.
  - Linguagem de comandos: particular de cada SO com sua estrutura e sintaxe.



# Componentes de um Sistema Operacional

## ▪ Serviços encontrados no núcleo

- Tratamento de interrupções e exceções
- Criação e eliminação de processos e threads
- Sincronização e comunicação entre processos e threads
- Escalonamento e controle dos processos e threads
- Gerência de memória
- Gerência do sistema de arquivos
- Gerência dos dispositivos de E/S
- Suporte a rede locais e distribuídas
- Contabilidade do uso do sistema
- Auditoria e segurança do sistema

# Componentes de um Sistema Operacional

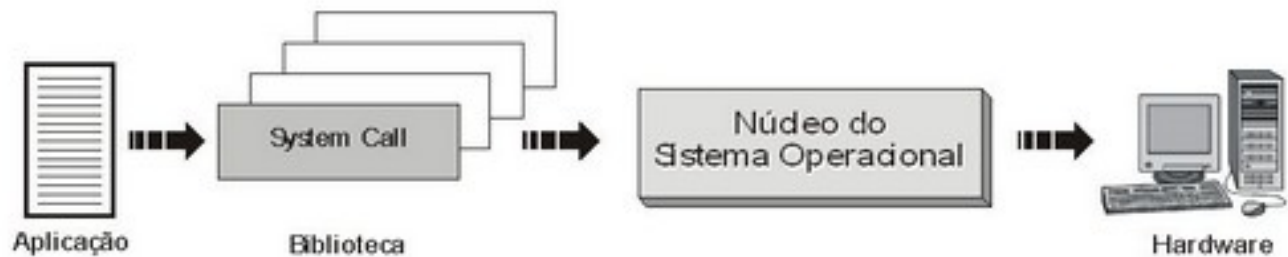
## ▪ Modos de acesso

- O sistema operacional é responsável por implementar uma proteção apropriada para o sistema e para isso é necessário o auxílio do processador (hardware).
- A forma usual é definir dois modos de operação:
  1. **Modo usuário:** algumas instruções não podem ser executadas.
  2. **Modo supervisor, núcleo ou kernel:** não existem restrições, e qualquer instrução pode ser executada. Essas instruções são chamadas de privilegiadas.
- Se um processo de usuário tentar executar uma instrução privilegiada, o hardware automaticamente gera uma interrupção e aciona o sistema operacional que poderá abortar o processo.

# Componentes de um Sistema Operacional

## ▪ System Calls

- Se uma aplicação precisa realizar alguma instrução privilegiada, ela realiza uma chamada ao sistema (system call), que altera do modo usuário para o modo kernel.
- Muitos dos serviços do SO são disponibilizados aos programadores através das *system calls*.
- Em computação, uma **chamada de sistema (system call)** é o mecanismo usado pelo programa para requisitar um serviço do sistema operacional, ou mais especificamente, do núcleo do sistema operacional.
- No Windows as *System Calls* são chamadas de API - Application Program Interface.
- As chamadas diferem de SO para SO, no entanto, os conceitos relacionados às chamadas são similares independentemente do SO.

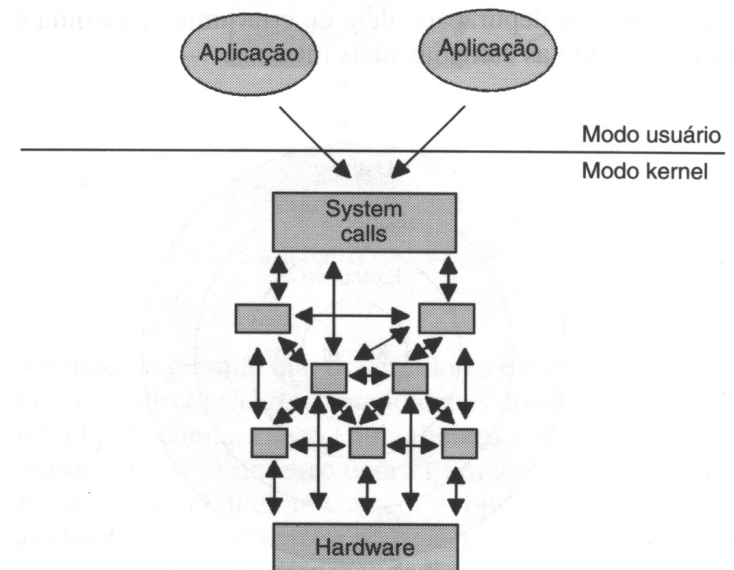


- Já que tivemos uma visão externa de um sistema operacional iremos visualizar sua estrutura interna.
- A arquitetura do sistema operacional pode ser classificada:
  - Arquitetura Monolítica
  - Arquitetura de Camadas
  - Máquina Virtual
  - Arquitetura Microkernel

# Arquitetura

## ▪ Arquitetura monolítica

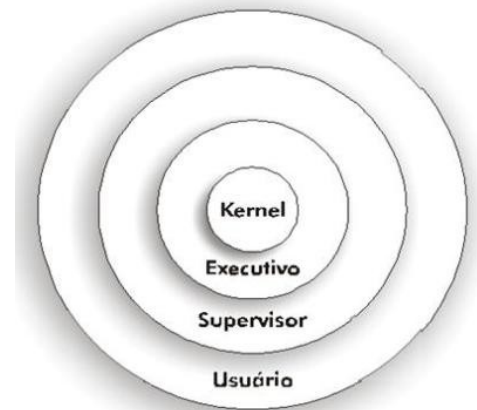
- O sistema operacional é escrito como um conjunto de procedimentos, sendo que cada um pode chamar um dos demais sempre que é necessário.
- **Construção:** compila-se todos os procedimentos com o objetivo de criar os módulos-objeto.
- **Prós:** desempenho.
- **Contra:** difícil manutenção.
- **Sistema que utilizavam:** DOS, primeiros UNIX.



# Arquitetura

## ▪ Arquitetura de camadas

- Com o aumento da complexidade e do tamanho do código dos sistemas operacionais, foram incorporadas técnicas de programação estruturada e modularizada.
- O sistema é dividido em níveis sobrepostos: cada camada oferece um conjunto de instruções que podem ser utilizadas pelas camadas superiores.
- **Prós:** o isolamento dos serviços do sistema operacional facilita a manutenção. A hierarquia protege as camadas mais internas.
- **Contras:** diminuição do desempenho por causa da quantidade de mudança de modos de acesso.
- **Sistema que a utilizam:** maioria das versões do Windows 2000 em diante, maioria dos Linux e Unix.



# Arquitetura

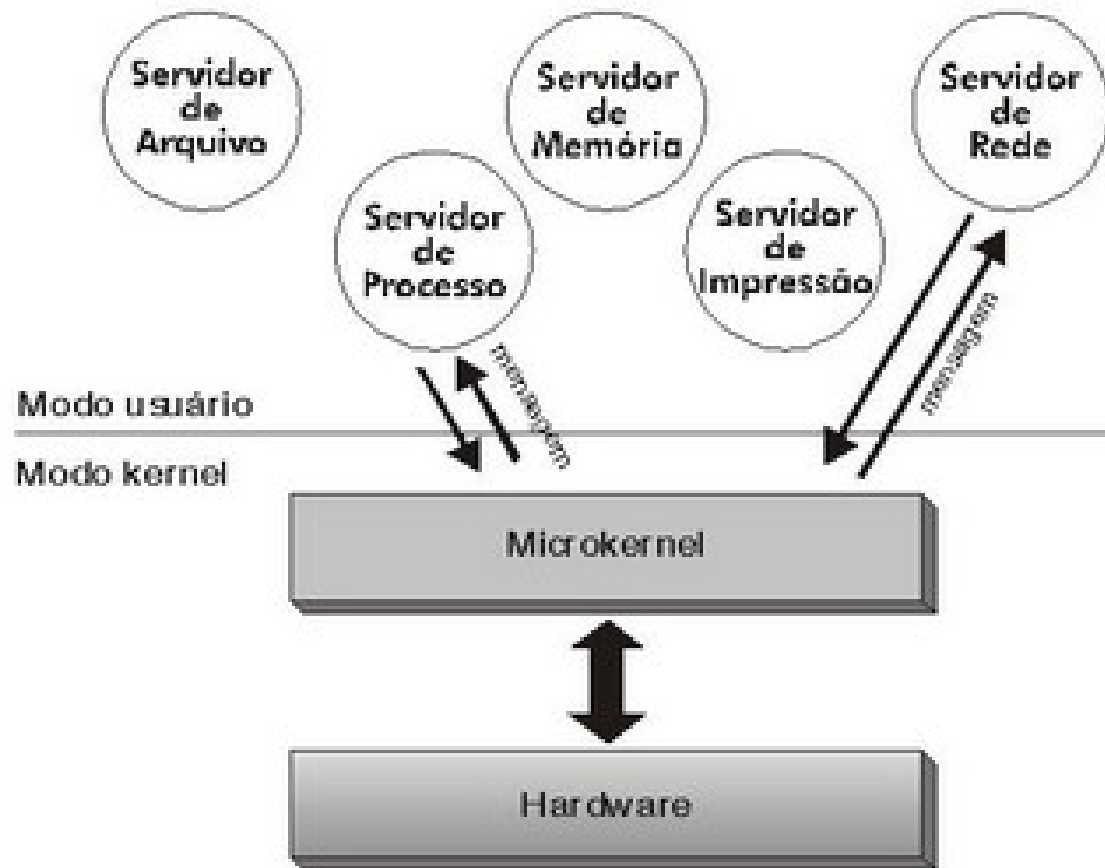
## ▪ Arquitetura Microkernel

- É fruto da tendência do núcleo de SO se tornar o menor possível. Os serviços são disponibilizados em processos. Cada processo é responsável em gerenciar um conjunto específico de funções como gerência de memória, gerência de arquivos, gerência de processos etc.;
- Dois tipos: **processo cliente** e **processo servidor**;
- A principal função do núcleo é gerenciar a comunicação entre esses processos.
- **Prós:** Maior proteção do núcleo: todos os processos são executados em modo usuário; Alta disponibilidade: se um servidor falhar, o sistema não ficará altamente comprometido; Maior eficiência: a comunicação entre serviços poderá ser realizada entre vários processadores ou até mesmo várias máquinas distribuídas;
- **Contra:** Grande complexidade para sua implementação; Menor desempenho devido à necessidade de mudança de modo de acesso.
- **Sistema que a utilizam:** Maioria das iniciativas ligadas ao desenvolvimento de SO distribuídos



# Arquitetura

## ▪ Arquitetura Microkernel



## ▪ Máquina Virtual

- O modelo de Máquina Virtual ou *Virtual Machine* (VM) cria um nível intermediário entre o *hardware* e o sistema operacional, denominado gerência de máquinas virtuais.
- Cria várias máquinas virtuais (VM) independentes onde cada uma oferece uma cópia virtual do hardware, incluindo os modos de acesso, interrupções, dispositivos de entrada/saída, etc.
- Cada VM é independente e é possível que cada VM possua seu próprio SO e que os usuários executem aplicações como se estivesse dedicado a cada um deles.
- **Prós:** Cria isolamento total de cada VM.
- **Contra:** Grande complexidade devido a necessidade de compartilhar e gerenciar os recursos de hardware entre as diversas VMs.

## ▪ Máquina Virtual

