

Sistemas Operacionais

Estruturas de SO

Edeyson Andrade Gomes

www.edeyson.com.br

Roteiro da Aula

- ▶ Estrutura do SO
- ▶ Chamadas ao Sistema
- ▶ Sistemas Monolíticos
- ▶ Sistemas em Camadas
- ▶ Sistemas Cliente-Servidor

Estrutura do SO

- ▶ **Sistema Operacional**
 - ▶ Conjunto de rotinas executadas sem ordem pré-estabelecida
 - ▶ Execução sob demanda (eventos assíncronos)
 - ▶ Atendimento ao hardware (Vetor de Interrupções)
 - ▶ Tarefas do próprio SO
 - ▶ Cada rotina oferece um serviço
 - ▶ Conjunto das rotinas
 - ▶ *Kernel* (Núcleo)

Estrutura do SO

- ▶ Principais Funções do *Kernel*
 - ▶ Tratamento de interrupções
 - ▶ Criação e eliminação de processos
 - ▶ Criar Processo:
 - Criar Contexto de SW e HW
 - Arquivos abertos são alocados no Contexto de SW
 - Alocar Memória
 - Executar o Loader

Estrutura do SO

- ▶ Principais Funções do *Kernel*
 - ▶ Eliminação de processos
 - ▶ Desalocar Contexto de SW e HW
 - ▶ Desalocar Memória
 - ▶ Desalocar recursos
 - Arquivos abertos e não fechados devem ser liberados
 - Liberar portas SOCKET

Estrutura do SO

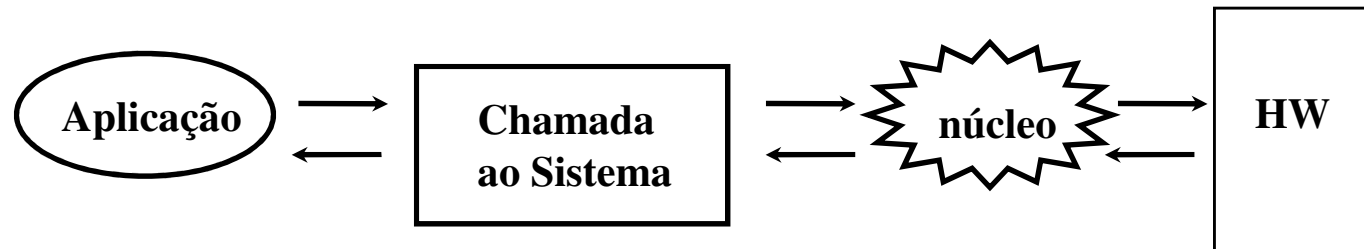
- ▶ Principais Funções do *Kernel*
 - ▶ Sincronização e comunicação entre processos
 - ▶ Escalonamento e controle de processos
 - ▶ Escolha de qual o próximo processo a executar
 - ▶ Gerência de memória
 - ▶ Alocar e desalocar espaço
 - ▶ Gerência do sistema de arquivos

Estrutura do SO

- ▶ **Principais Funções do *Kernel***
 - ▶ Operações de E/S
 - ▶ Contabilização do uso do sistema
 - ▶ Tempo de CPU, consumo de recursos, etc.
 - ▶ Segurança do sistema

- ▶ **O Sistema Operacional deve:**
 - ▶ Oferecer serviços
 - ▶ Proteger o núcleo

Chamadas ao Sistema



- ▶ `app1.c => fread(...)`
 - ▶ `app2.pas => read(...)`
 - ▶ `app3.bas => input(...)`
- EsSES comandos fazem a leitura do disco
São chamadas a procedimento

Os compiladores devem traduzir as chamadas a procedimento em chamadas ao sistema.

Logo, `fread`, `read` e `input` serão traduzidas para a chamada `READ(...)` que o SO entende. Isso é uma Chamada ao Sistema.

Chamadas ao Sistema

- ▶ Mecanismo para oferecer serviços
- ▶ Aplicações fazem chamadas às rotinas do SO
 - ▶ *System Calls*
 - ▶ Parâmetros de entrada
 - ▶ Retorno de valores
 - ▶ Semelhante a procedimentos de programação alto nível
 - ▶ Cada serviço do S.O. tem sua rotina (*system call*)
 - ▶ Biblioteca de Chamadas ao Sistema (*system calls*)

Chamadas ao Sistema

◆ **Divisão das *System Calls* por Função**

- Gerência de Processos
 - » Criação e eliminação de processos
 - » Alteração das características do processo
 - » Prioridade
 - » Sincronização e comunicação entre processos
- Gerência de Memória
 - » Alocação e liberação de memória
- Gerência de E/S
 - » Operações de E/S
 - » Manipulação de arquivos e diretórios

Estrutura do SO

- ▶ Mecanismo para proteção do sistema
- ▶ Algumas instruções podem afetar a integridade do sistema
 - ▶ Acesso aos arquivos (instruções E/S)
 - ▶ Acesso à memória
- ▶ Instruções de Máquina
 - ▶ Privilegiadas x Não Privilegiadas
- ▶ Modo de Acesso Usuário
 - ▶ Somente instruções não-privilegiadas
 - ▶ Executado por aplicações do usuário
 - ▶ Chamadas de rotinas do sistema (*system calls*)

Estrutura do SO

- ▶ **Modo de Acesso *Kernel***
 - ▶ Todas instruções do processador
 - ▶ Executado apenas o SO (núcleo)
- ▶ **Controle do Modo de Acesso**
 - ▶ Via HW, através de registrador da CPU
 - ▶ Tentativa de violação gera exceção

Privilégio de Instruções

- ▶ Exemplo do Hotel
 - ▶ Gerente e Hóspedes

Estrutura do SO

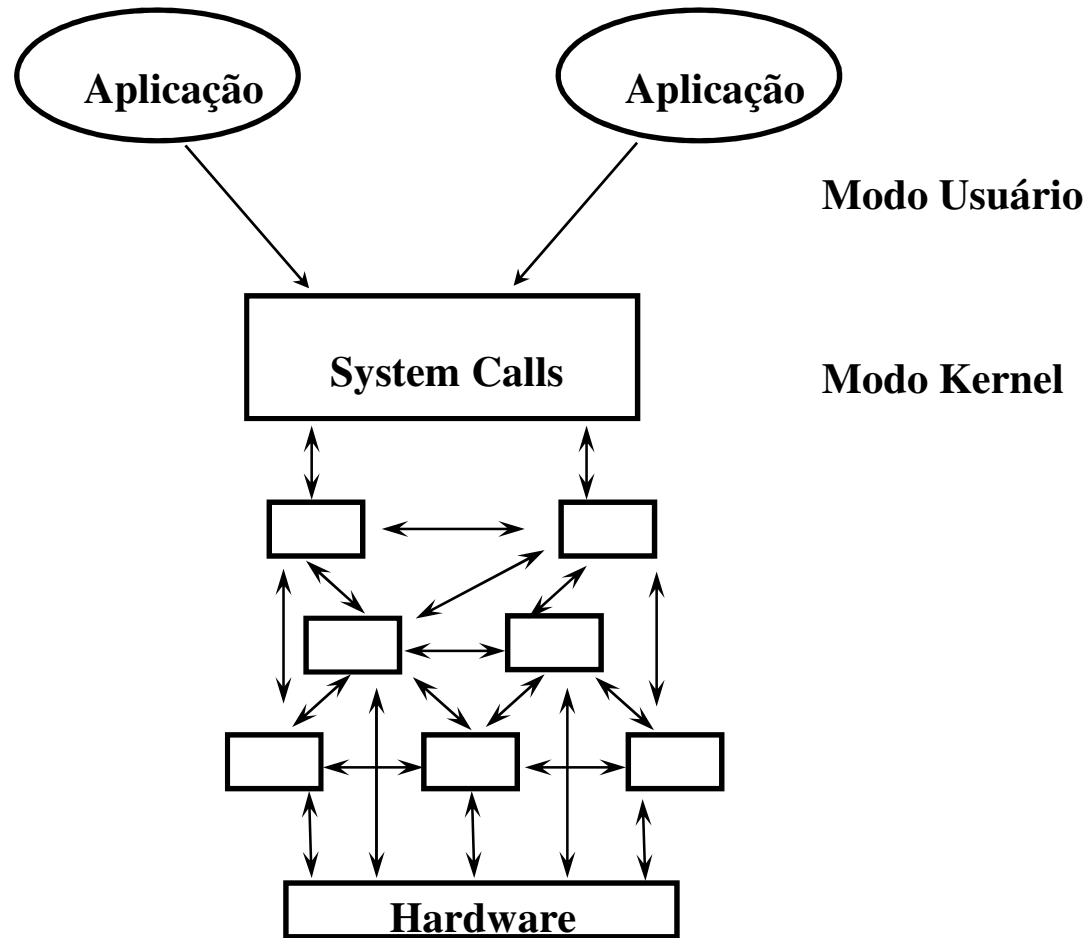
- ▶ Organização do código do SO
 - ▶ Relacionamento entre os componentes
 - ▶ Sistemas Monolíticos
 - ▶ Em Camadas
 - ▶ Cliente Servidor

Sistemas Monolíticos

- ▶ **Sistemas Monolíticos**

- ▶ Conjunto de rotinas que interagem livremente
 - ▶ Acoplamento forte
 - ▶ Alto desempenho
 - ▶ Ruim de manter e alterar o código
- ▶ Análogo a aplicação com vários módulos separados e ligados para formar um executável

Sistemas Monolíticos



Sistemas em Camadas

▶ Sistemas em Camadas

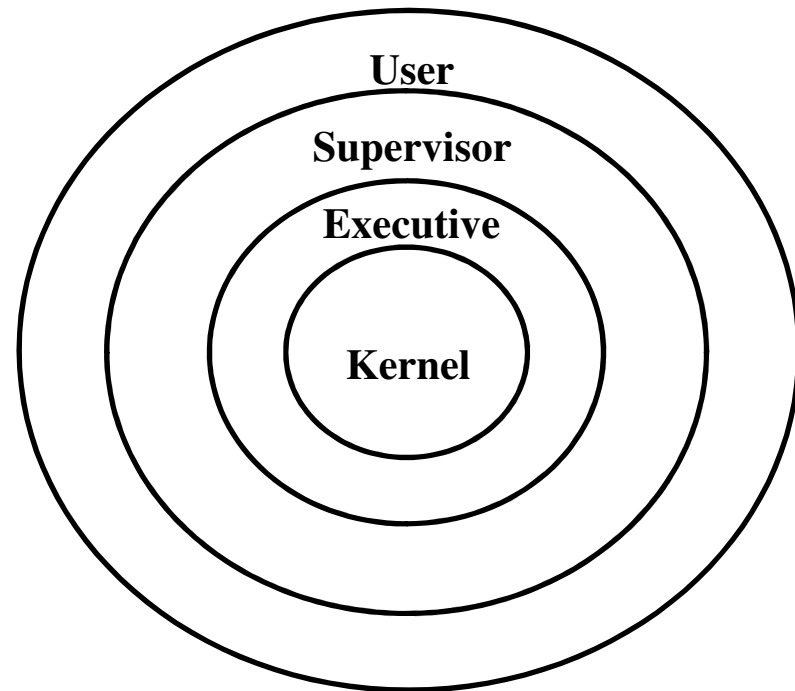
- ▶ Camadas sobrepostas
- ▶ Cada camada oferece serviços para a camada superior
- ▶ Módulos de uma camada podem fazer referência apenas aos módulos das camadas inferiores
- ▶ Camadas inferiores são mais privilegiadas
- ▶ Estruturação do SO
 - ▶ Isolamento de funções
 - ▶ Facilidade de depuração / manutenção
 - ▶ Hierarquia de níveis

Sistemas em Camadas

Sistema MULTICS

5	Operador
4	Programas de Usuário
3	Entrada / Saída
2	Comunicação
1	Gerência de Memória
0	Multiprogramação

Sistema VMS



Sistemas Cliente-Servidor

- ▶ **Sistemas Cliente-Servidor**
 - ▶ Núcleo menor e mais simples
 - ▶ Divisão em processos
 - ▶ Serviço de arquivos
 - ▶ Serviço de processos
 - ▶ Serviço de memória
 - ▶ Modelo Cliente / Servidor
 - ▶ Aplicação (cliente) / Processo do SO (Servidor)
 - ▶ Comunicação entre Cliente / Servidor através de Mensagens

Sistemas Cliente-Servidor

- ▶ Núcleo é responsável pela troca de mensagens
- ▶ Estruturação do SO
 - ▶ Isolamento de funções por servidores específicos
 - ▶ Facilidade de depuração e manutenção
- ▶ Possibilidade de uso de Sistemas Multiprocessados
 - ▶ Fortemente ou Fracamente Acoplados

Sistemas Cliente-Servidor
