

SISTEMAS OPERACIONAIS

Introdução a Processos

Andreza Leite
andreza.leite@univasf.edu.br

Plano da Aula

2

- Introdução a Processos
- Threads vs Processos
- Características dos Processos
- Estados dos Processos

Processos

3

□ **Processos/Multiprogramação**

- Por que executar vários programas simultaneamente?
 - Permitir que vários usuários usem uma máquina simultaneamente
 - Melhorar a eficiência do sistema

□ **Processo**

- Um programa em execução

Processos



- Em sistema operacional é conveniente diferenciar um programa de sua execução:
 - ▣ Programa - entidade estática e permanente
 - composto por uma seqüência de instruções: passivo sob o ponto de vista do sistema operacional
 - ▣ Processo - entidade dinâmica
 - altera seu estado a medida que avança sua execução;
 - o processo é uma abstração que representa um programa em execução;

Processos

5

- Um processo é composto por:
 - Programas
 - Dados
 - Contexto (valores)

Processos

6

- Analogia entre um **Processo** e um **Cozinheiro**
- Imagine um engenheiro com dotes culinários fazendo um bolo:
 - ▣ Receita = programa
 - ▣ Engenheiro cozinheiro = processador (CPU)
 - ▣ Ingredientes = dados de entrada.
 - ▣ Processo é a atividade desempenhada pelo cozinheiro em ler a receita, buscar os ingredientes e assar o bolo.



Processos

7

- **Características/Propriedades de um processo**
 - ▣ Um processo é uma abstração que representa um programa em execução
 - Um processo tem execução sequencial
 - O resultado da execução de um processo independe da velocidade do processador em que for executado

- O mesmo programa executado por dois usuários gera dois processos

Processos

8

- **Características/Propriedades de um processo**
 - Um programa pode gerar (criar) vários processos
 - Um processo tem duas partes:
 - Ativa - fluxo de controle
 - Passiva - espaço de endereçamento (memória, registradores, arquivos)

Processos

9

□ Thread

- Denota um fluxo de controle (Processo leve)
- Por questão de eficiência, processos podem ter múltiplas threads que compartilham o espaço de endereçamento do processo

□ Escalonador

- Programa que controla/decide que thread deve ser executada a cada instante

Processos

10

□ Ciclos de um processo

□ Criados

- Início do sistema
- Chamada de Sistema de criação de processo por um processo em execução
- Requisição do usuário para criar um novo processo
- Início de uma tarefa em lote (computadores de grande porte)

□ Destruídos

- Saída normal (voluntária)
- Saída por erro (voluntária)
- Erro fatal (involuntário)
- Cancelamento por outros processos (involuntária)

Processos

11

□ Ciclos básicos de operação

□ Ciclo de processador

- Tempo que ocupa a CPU

□ Ciclo de Entrada/Saída

- Tempo de espera de um evento E/S

□ A troca de ciclos ocorre por:

□ Chamada de Sistema (CPU → E/S)

□ Ocorrência de Evento - Interrupção (E/S → CPU)

- inserir um pendrive na USB, escrever um bloco em disco, receber um pacote pela rede...

Processos

12

- **Processos CPU-bound**

- Ciclo de processador $>$ Ciclo de E/S

- **Processos I/O bound**

- Ciclo de E/S $>$ Ciclo de processador

Processos

13

□ Relacionamentos:

□ Processos Independentes

- Não apresentam relacionamentos com outros processos

□ Grupo de Processos

- Apresentam algum tipo de relacionamento , por exemplo, hierarquia de processos com mesma filiação:
 - Processo criador é o processo pai
 - Processo criado é o processo filho
 - Representação através de uma árvore
 - Evolução dinâmica

Processos

14

□ Relacionamentos:

□ Processos Independentes

- Não apresentam relacionamentos com outros processos

□ Grupo de Processos

- Apresentam algum tipo de relacionamento , por exemplo, hierarquia de processos com mesma filiação:
 - O que fazer na destruição de um processo?
 - Toda descendência “morre”?
 - A descendência é herdada pelo processo “avô”?
 - Postergar a destruição efetiva do processo pai até o final de todos os filhos?

Processos

15

□ ESTADOS

- Após ser criado, um processo precisa entrar num ciclo de processador

- Possibilidades:
 - Processador não está disponível
 - Vários processos estão sendo criados

- O que fazer então?
 - Criar uma fila de processos prontos para executar

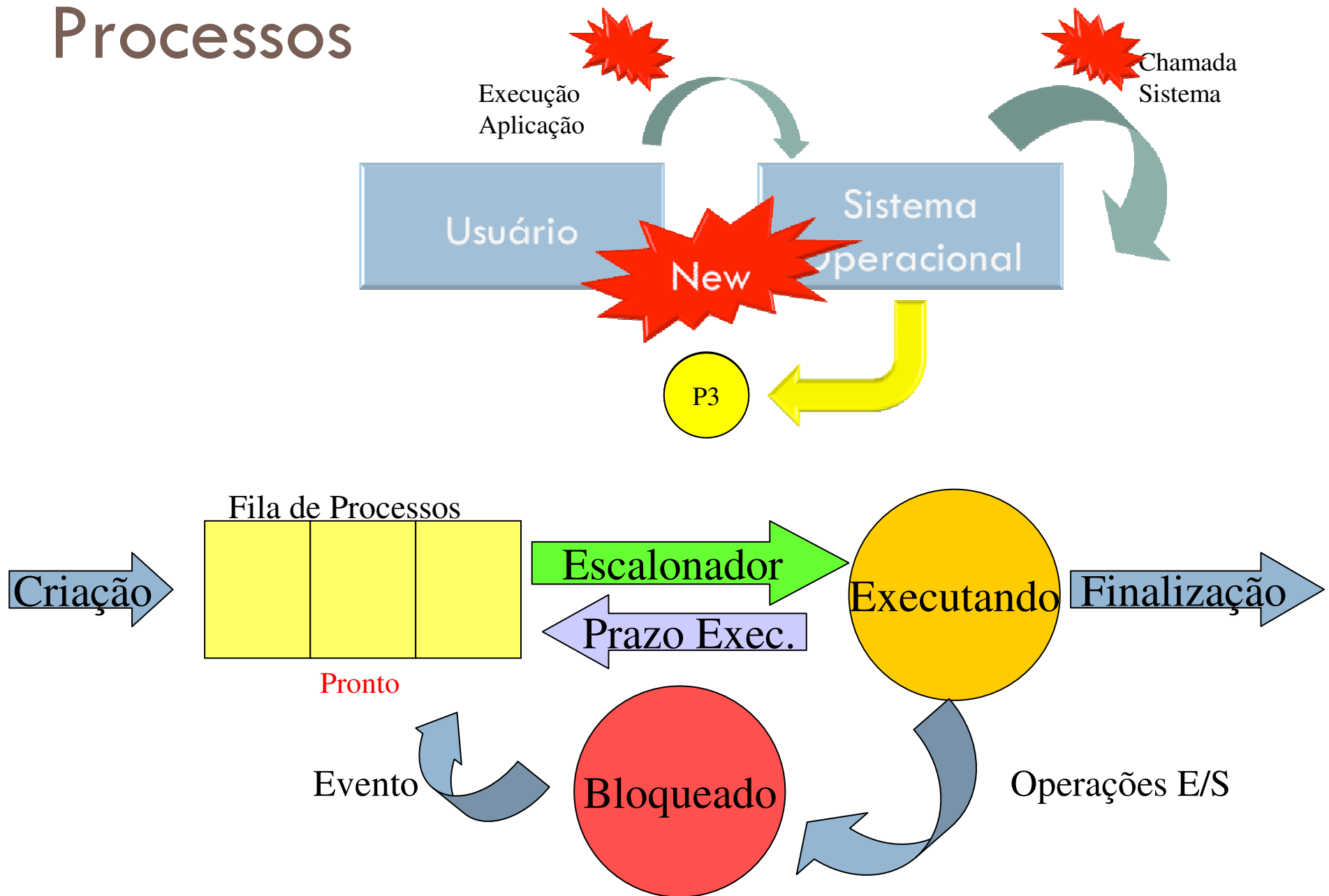
Processos

16

□ ESTADOS

- Diagrama de transição de estados de um processo
- Um escalonador, através do uso da **fila de processos prontos** (aptos para a execução) atribui a cada instante, o **processador** a um dos **processos**

Processos



Processos

18

□ ESTADOS

□ Finalização de processos

- Final normal da execução
- Excesso de tempo
- Falta de memória
- Violação de limites
- Erros de proteção
- Erros aritméticos
- Timeout de eventos
- Erro em periféricos de E/S
- Intervenção do SO
- Logoff de usuários

Processos

19

□ Escalonador

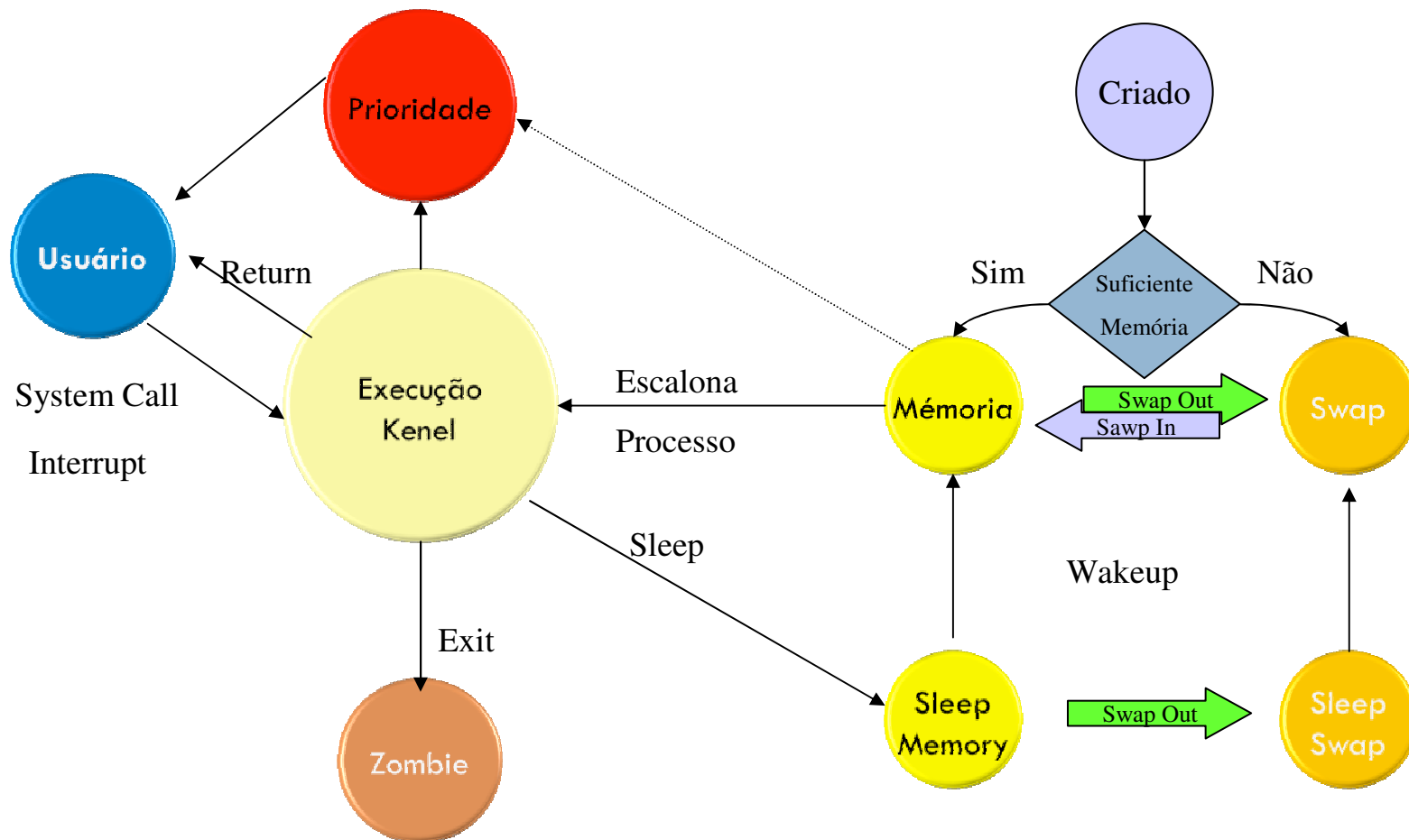
□ Cada processo é representado por um PCB (Process Control Block) que contém:

- Estado do processo
 - Rodando (Executando)
 - Pronto
 - Bloqueado
- Número do processo
- Contador de Programa
- Registradores
- Localização da pilha de execução
- Prioridade de execução

Processos

20

□ Escalonador



Processos

21

□ Resumindo:

- ▣ Um processo é uma abstração de um programa em execução
- ▣ Funções do S.O.
 - Alocar recursos a processos
 - Suportar criação de processos pelo usuário
 - Suportar comunicação entre processos

É função do Escalonador de Processos:

Dividir tempo de CPU para diferentes processos de forma a maximizar a utilização da CPU, fornecendo um tempo de resposta razoável!

Processos

22

□ Resumindo:

□ Criação de um processo:

- O que é necessário?
 - construir estruturas de dados
 - alocar espaço de endereçamento

□ Quando?

- usuário inicia sessão;
- quando gerado por outro processo
 - (e.g., servidor de web ,ftp);

Processos

23

□ Resumindo:

□ Destruição de um processo:

■ Quando terminar?

- execução de instrução de finalização;
- condições de erro;

■ Algumas razões para o término:

- tempo excedido;
- falta de memória;
- uso de instrução privilegiada;
- término do processo pai; ...

Processos

24

□ Resumindo:

□ Estados de um Processo

■ Pronto (Ready)

- Quando um processo aguarda que o sistema operacional aloque a CPU para sua execução

■ Espera (Wait)

- Quando um processo está aguardando algum evento externo para prosseguir com o processamento

■ Executando (Running)

- Quando um processo está sendo processado pela CPU

■ Bloqueado

- Esperando operações de entrada ou saída