



Laboratório de Pesquisa em Redes e Multimídia

Processos

Estruturas de Controle



Universidade Federal do Espírito Santo
Departamento de Informática

Processos e Recursos (1)

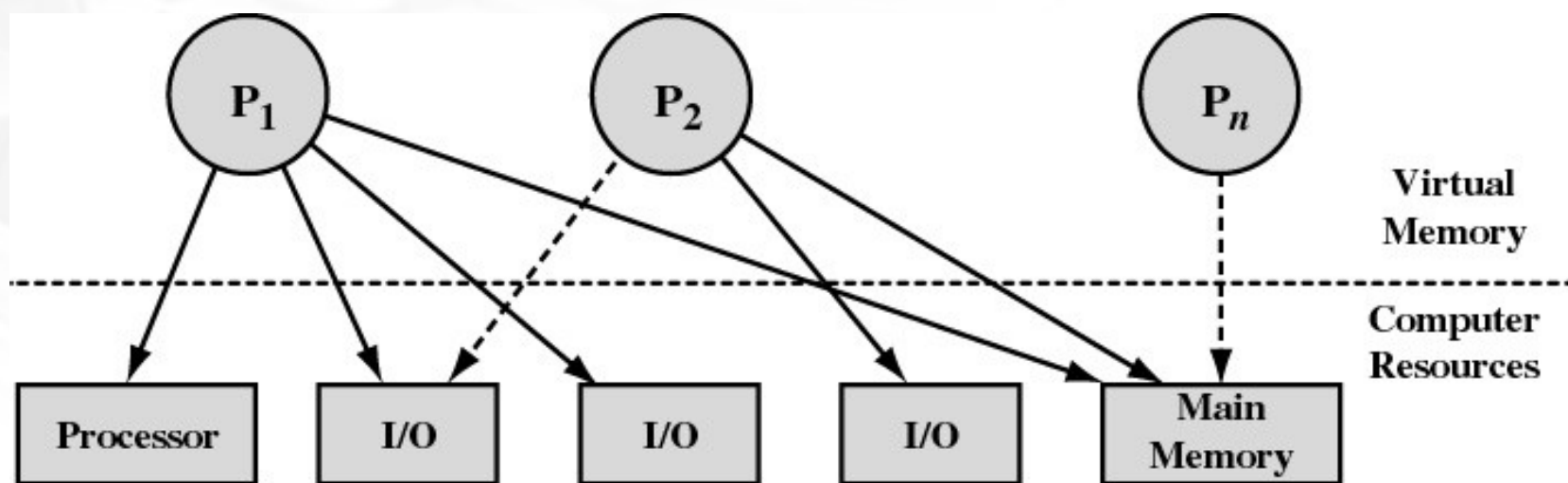


Figure 3.9 Processes and Resources (resource allocation at one snapshot in time)

Processos e Recursos (2)

- O S.O. gerencia recursos computacionais em benefício dos diversos processos que executam no sistema.
- A questão fundamental é:
 - Que **informações** o sistema operacional precisa manter para poder **controlar** os processos e **gerenciar** os recursos em benefícios deles?



Tabelas de Controle do S.O.

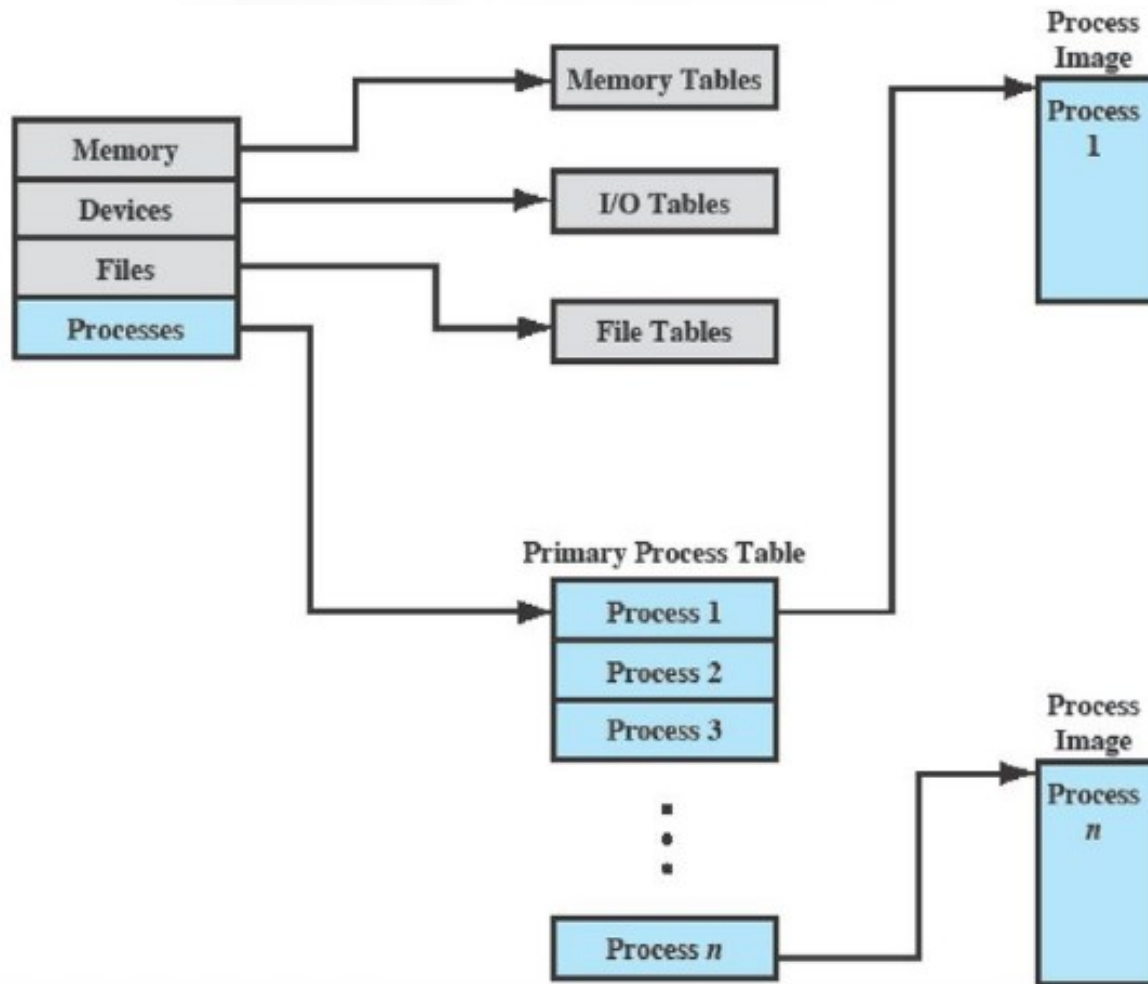


Imagem do Processo

- Elementos típicos da Imagem do processo

User Data

The modifiable part of the user space. May include program data, a user stack area, and programs that may be modified.

User Program

The program to be executed.

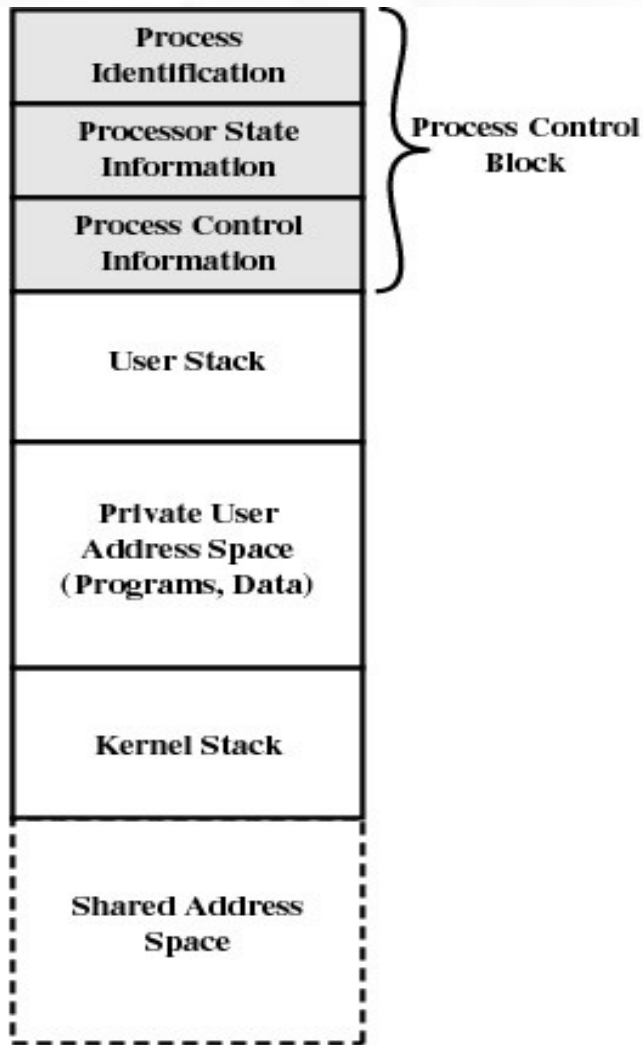
Stack

Each process has one or more last-in-first-out (LIFO) stacks associated with it. A stack is used to store parameters and calling addresses for procedure and system calls.

Process Control Block

Data needed by the OS to control the process.

Imagem do Processo



▪ PCB - Process Control Block

- ou Bloco de Controle de Processo

- Todas as informações que o S.O. precisa para poder controlar a execução do processo (atributos do processo)

- **Identificação**

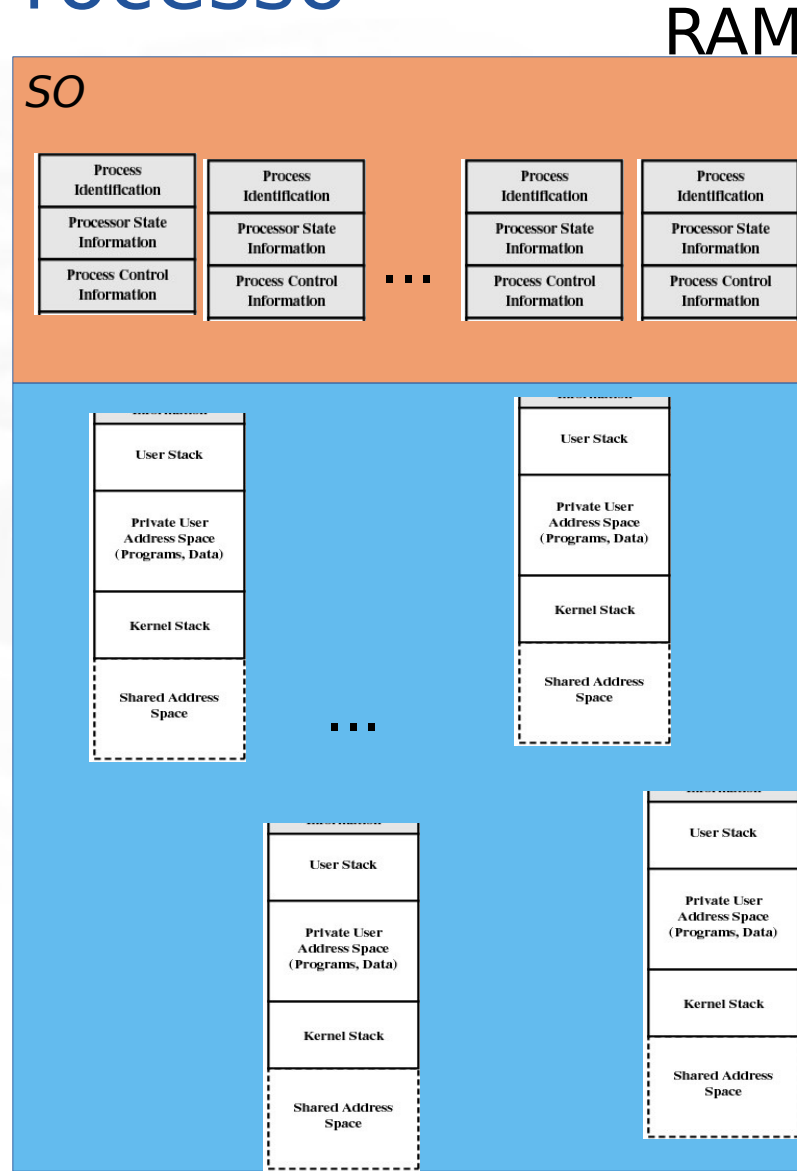
- **Contexto de Execução**

- **Outras infos de controle**

- Número fixo ou variável de PCBs no SO

- Alocação estática ou dinâmica

Imagem do Processo



PCB - Process Control Block

▪ Identificação do processo

- ID do processo, do processo pai, do usuário...

▪ Informações de Contexto de execução

- Registradores visíveis ao usuário
- Reg. de controle/estado: PC, SP, Flags, *Status* (modo supervisor /usuário, interrupção habilitada /desabilitada)...

▪ Informações de Controle do Processo

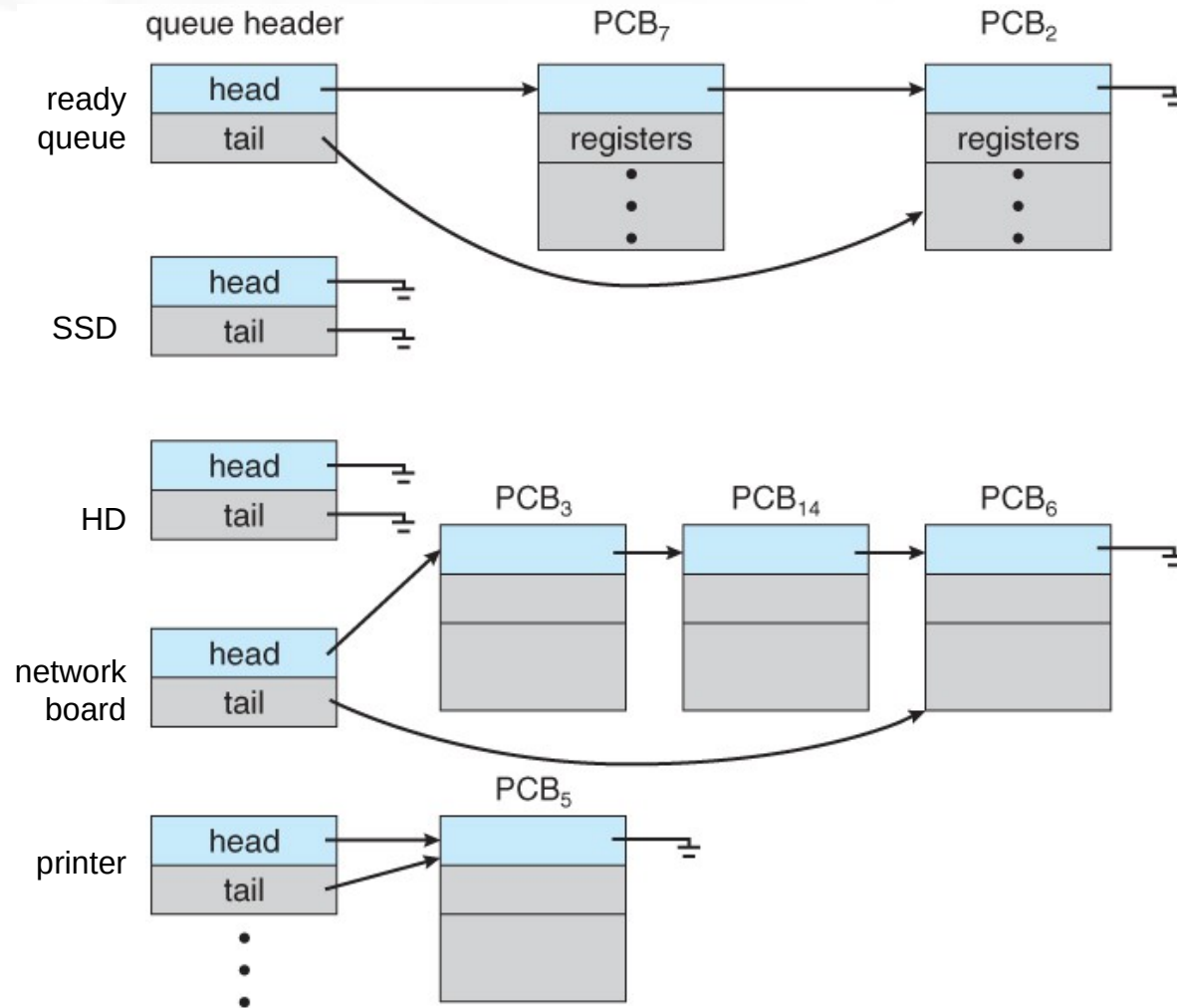
- ...

PCB - Process Control Block

... Informações de Controle do Processo

- Informações de **Escalonamento e Estado**:
 - Estado do processo (*ready, running, suspended, etc.*)
 - Prioridade (*default, corrente, máxima*)
 - Tempos de fila, execução em CPU, etc.
 - Evento que o processo está aguardando
- **Comunicação entre processos**:
 - Flags, sinais e mensagens podem estar associados com a comunicação entre dois processos independentes.
- **Privilégios** (memória, instruções, etc.)
- **Ownership** (ex: arquivos abertos)
- Estruturação de dados (ex: **ponteiros**)

PCBs e as Filas do Sistema



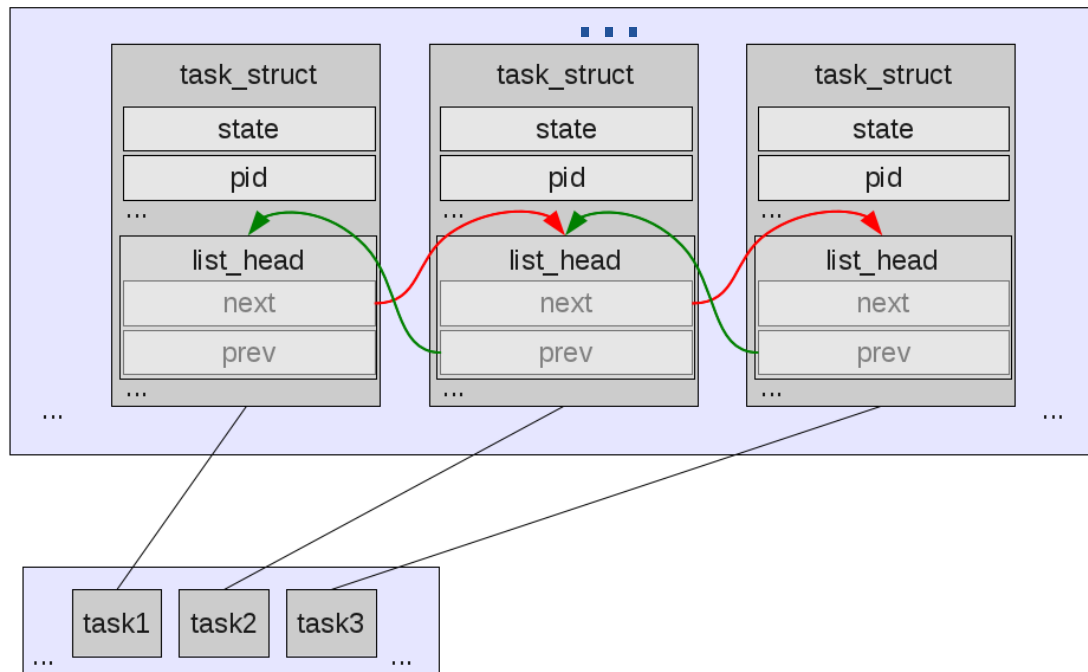
Informações de Controle do Processo (2)

- **Privilégios** em termos de **memória** que pode ser acessada, **instruções** que podem ser executadas, ou mesmo serviços e utilitários do sistema.
- **Gerência de Memória:**
 - Ponteiros para tabelas de páginas/segmentos que descrevem a memória virtual assinalada ao processo.
- **Ownership** e utilização de recursos:
 - Arquivos abertos;
 - Histórico de uso da UCP ou de outro recurso (para usos do escalonador);

Exemplo... PCB no Linux: Estrutura C task_struct

```
pid_t pid; /* process identifier */
long state; /* state of the process */
unsigned int time_slice /* scheduling information */
struct task_struct *parent; /* this process's parent */
struct list_head children; /* this process's children */
struct files_struct *files; /* list of open files */
struct mm_struct *mm; /* address space of this process */
```

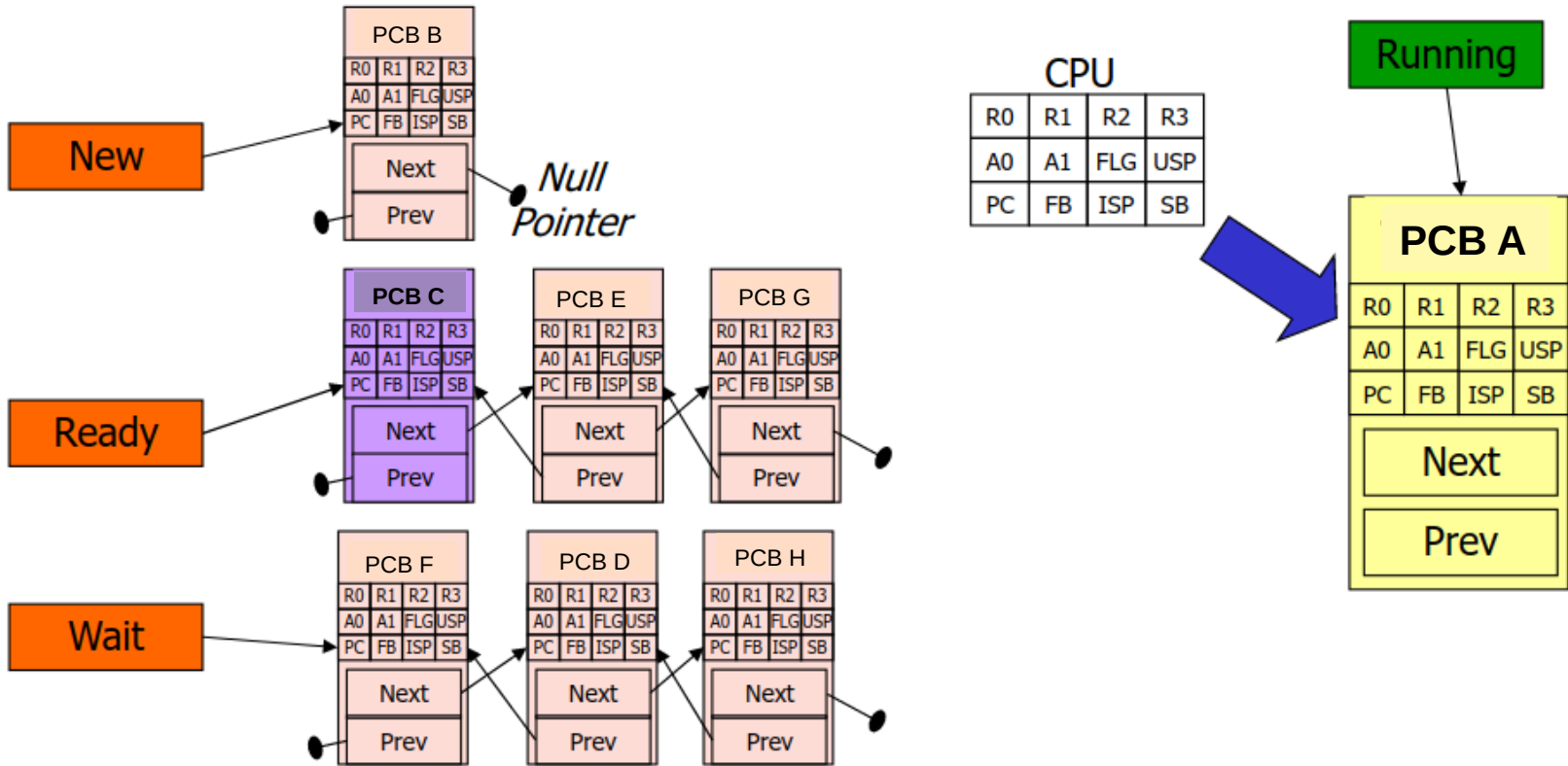
...



Troca de Contexto ⁽¹⁾

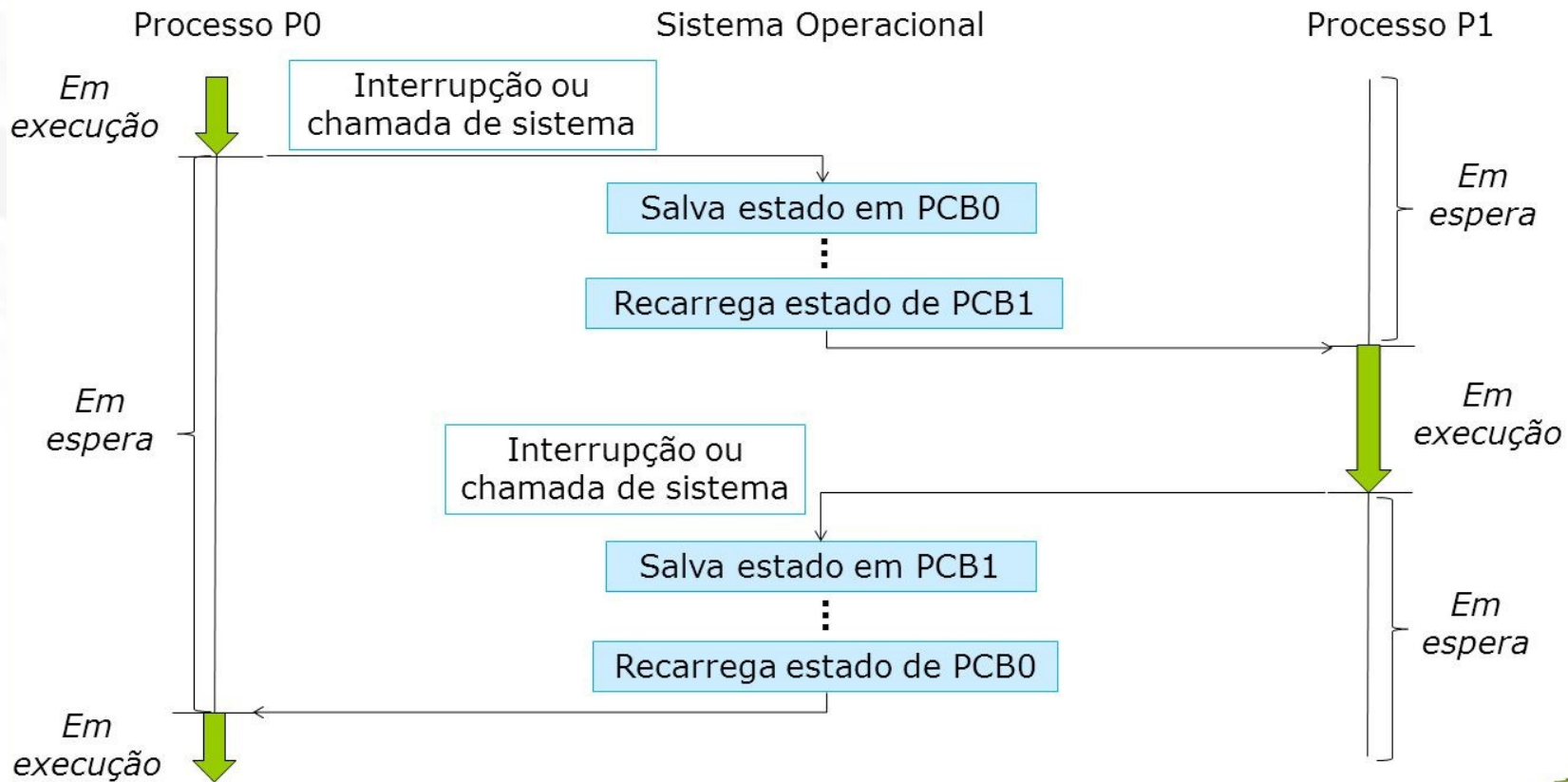
- **Contexto de execução:** estado do processador
 - Valores nos registradores
- Ações na troca de contexto
 - 1. Salvar o contexto** do processador, incluindo o PC e outros registradores.
 - 2. Alterar o PCB** do processo que estava no estado “em-execução” (*running*)
 - 1.- (Por exemplo, ele pode estar indo pra “Blocked”)
 - 3. Mover o PCB** para a fila apropriada.
 - 4. Selecionar outro processo** para execução.
 - 5. Alterar o PCB** do processo selecionado.
 - 6. Alterar as tabelas** de gerência de **memória**.
 - 7. Restaurar o contexto do processo selecionado.**

Ações na Troca de Contexto (2)





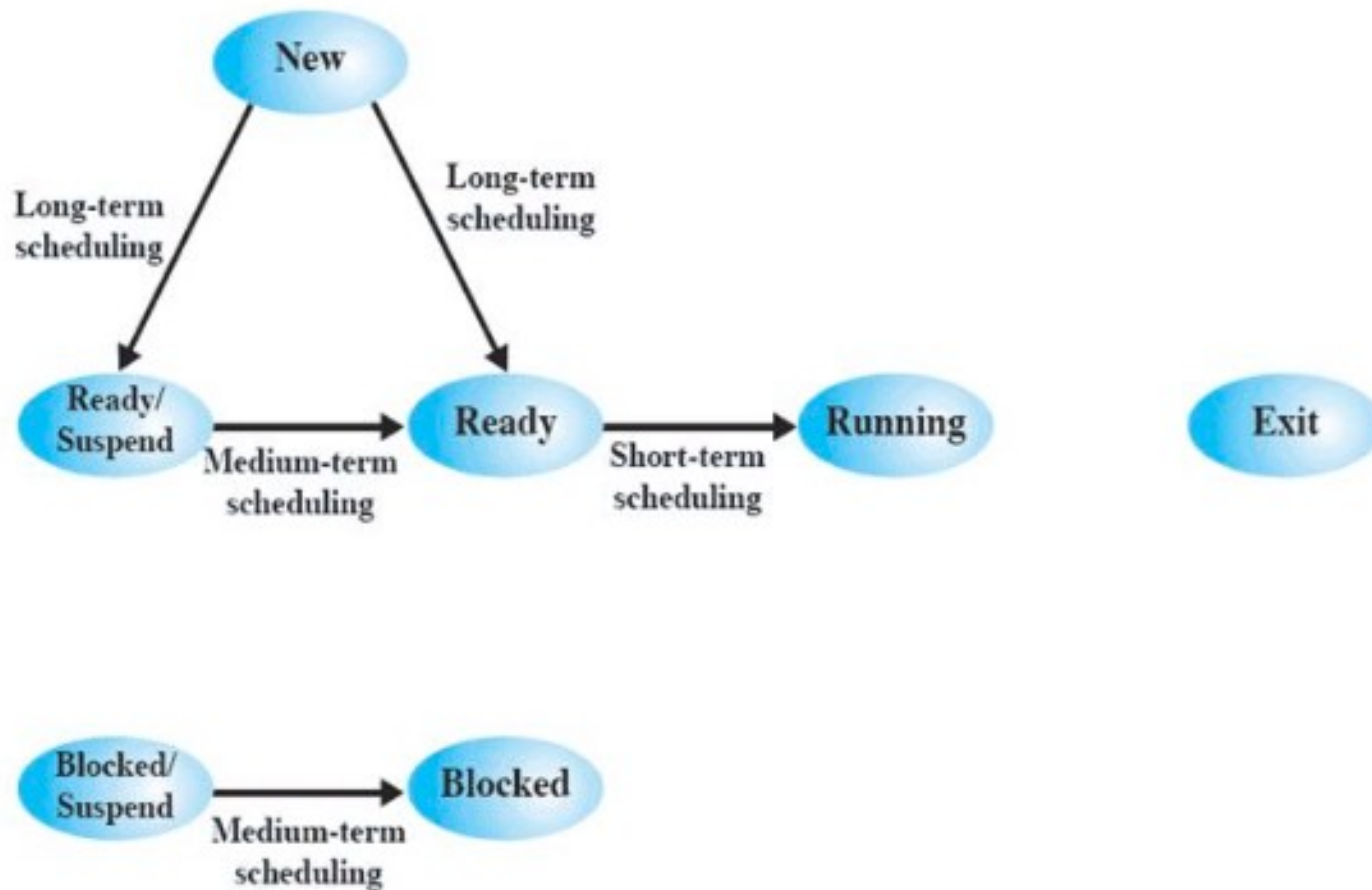
Troca de contexto



O Escalonador (“Scheduler”)

- Módulo do S.O. responsável pelo controle do recurso “processador”.
- Divide o tempo da UCP entre os processos do sistema.
- Três tipos básicos:
 - Escalonador de curto prazo (“*short-term scheduler*”);
 - Escalonador de longo prazo (“*long-term scheduler*”);
 - Escalonador de médio prazo (“*medium-term scheduler*”).

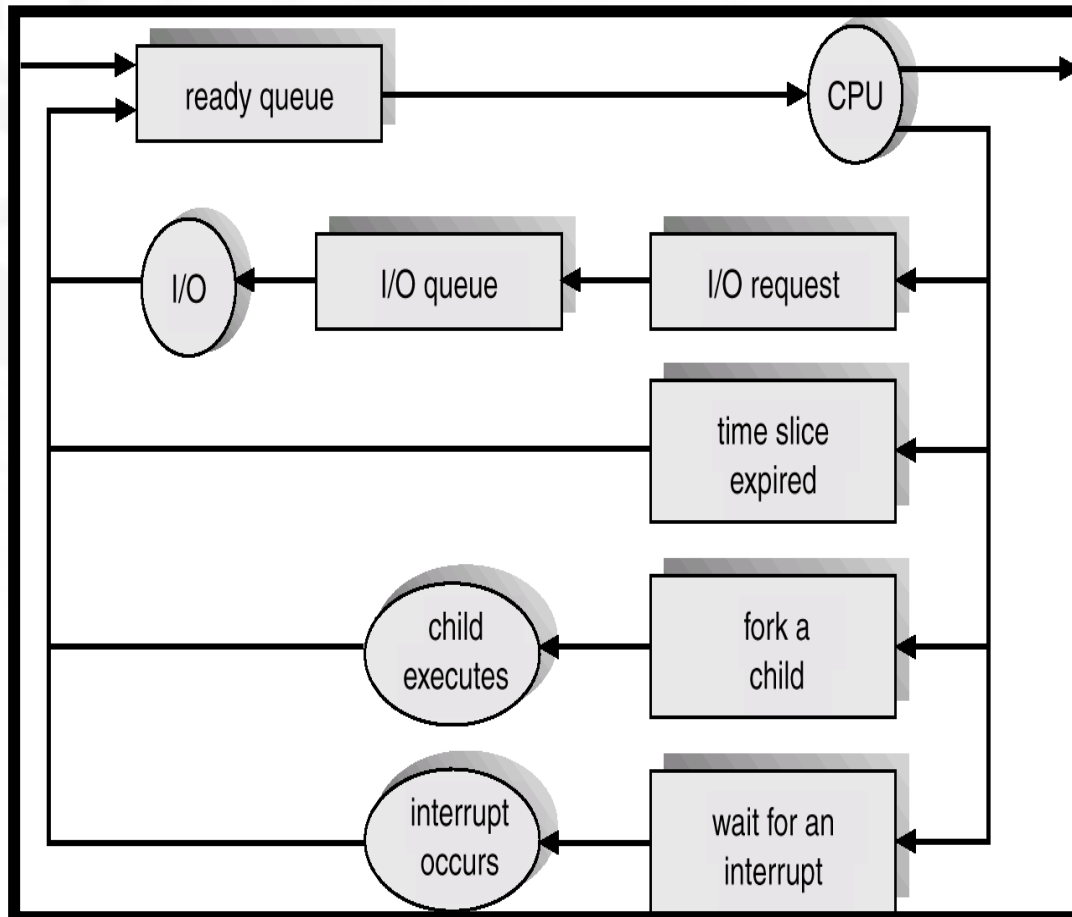
Escalonamento e a Transição de Estados



Escalonador de Curto Prazo (1)

- Escalonador da UCP
 - ***Dispatcher***, *CPU Scheduler*
- Seleciona qual processo deve ser executado a seguir (*ready*→*running*).
- É invocado **muito freqüentemente** (ordem de milisegundos).
 - Deve, portanto, ser rápido.

Escalonador de Curto Prazo (2)



Escalonador de Longo Prazo

- Escalonador de *Jobs* (“*Job Scheduler*”).
- Seleciona quais processos devem ser levados para a fila de prontos (*new*→*ready*).
- Baixa frequência de invocação (ordem de segundos ou minutos).
- Permite o controle da carga no sistema, (controla o grau de multiprogramação).

Escalonador de Médio Prazo

- Utiliza a técnica de *swapping*.
 - **Swap out:** a execução do processo é suspensa e o seu código e dados são temporariamente copiados para o disco.
 - **Swap in:** o processo é copiado de volta do disco para a memória e sua execução é retomada do ponto onde parou.
- Está intimamente ligado à **gerência de memória**.

Escalonamento e as Filas do Sistema

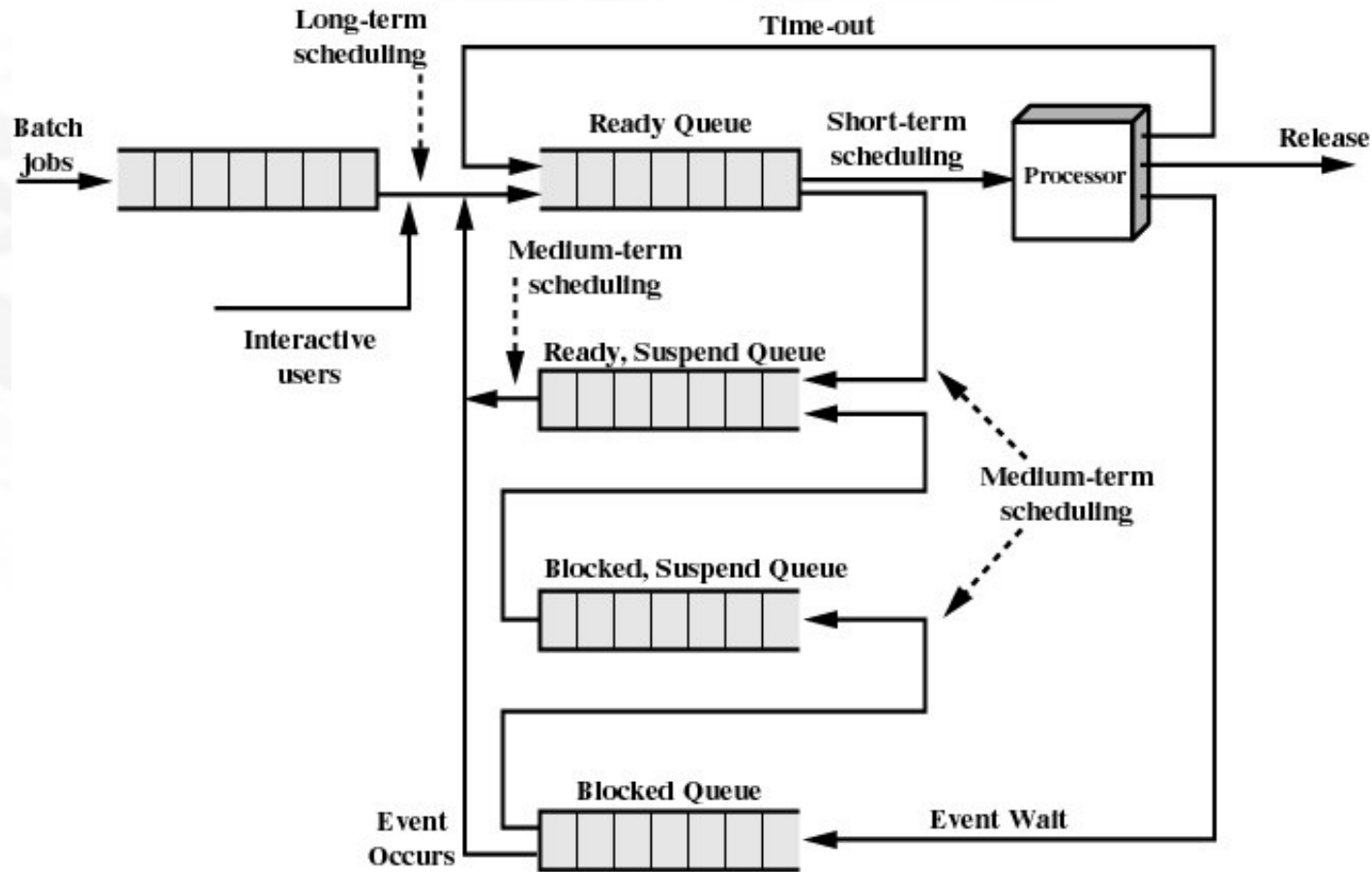


Figure 9.3 Queuing Diagram for Scheduling

Tipos de Escalonadores (Resumo)

Long-term scheduling	The decision to add to the pool of processes to be executed
Medium-term scheduling	The decision to add to the number of processes that are partially or fully in main memory
Short-term scheduling	The decision as to which available process will be executed by the processor
I/O scheduling	The decision as to which process's pending I/O request shall be handled by an available I/O device

Ciclos de CPU e de I/O (1)

load store
add store
read from file

} cpu burst

wait for I/O

} I/O burst

store increment
index
write to file

} cpu burst

wait for I/O

} I/O burst

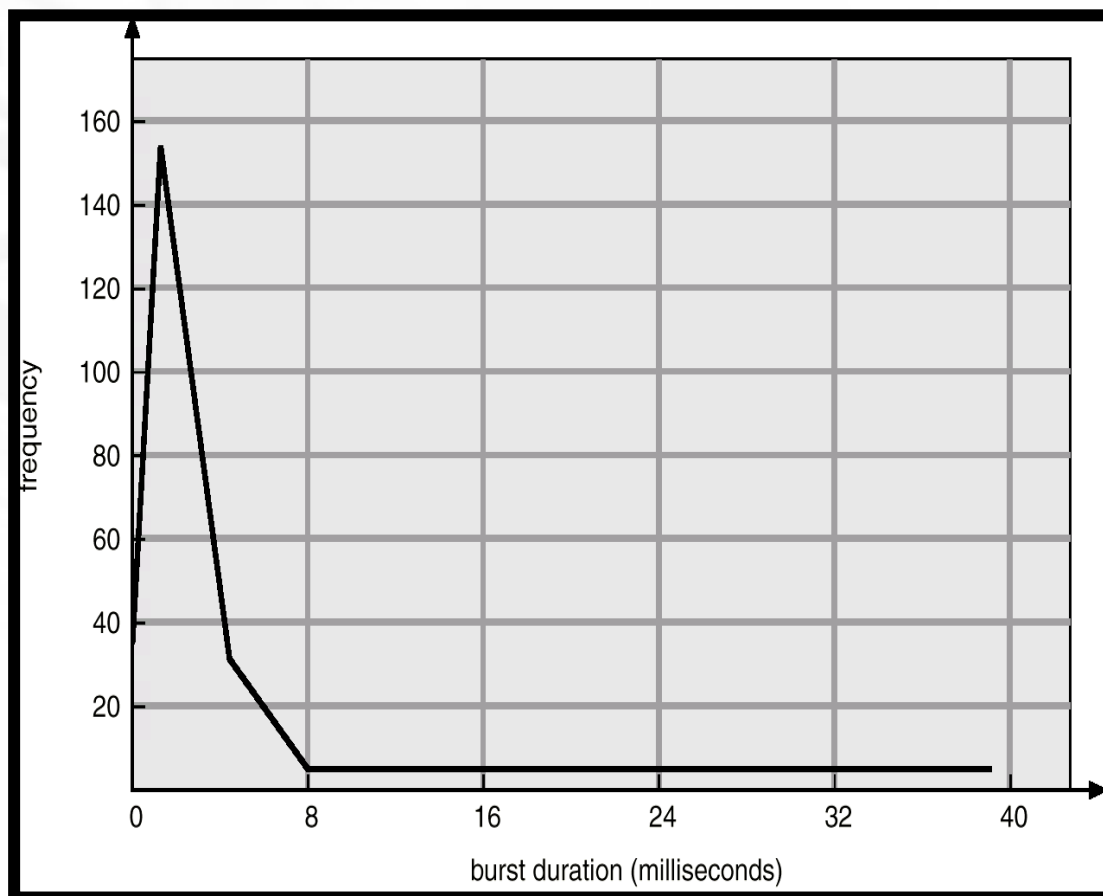
load store
add store
read from file

} CPU burst

wait for I/O

} I/O burst

Ciclos de CPU e de I/O (2)



Tipos de Processos

- a) Processo CPU Bound:
 - Uso intensivo de CPU.
 - Realiza pouca operação de E/S.
 - Pode monopolizar a CPU, dependendo do algoritmo de escalonamento.
- b) Processo I/O Bound:
 - Orientado a I/O.
 - Normalmente, devolve deliberadamente o controle da CPU

