

Curso: Engenharia de Computação

Arquitetura de Computadores

Prof. Clayton J A Silva, MSc
clayton.silva@professores.ibmec.edu.br



Nível do Sistema Operacional

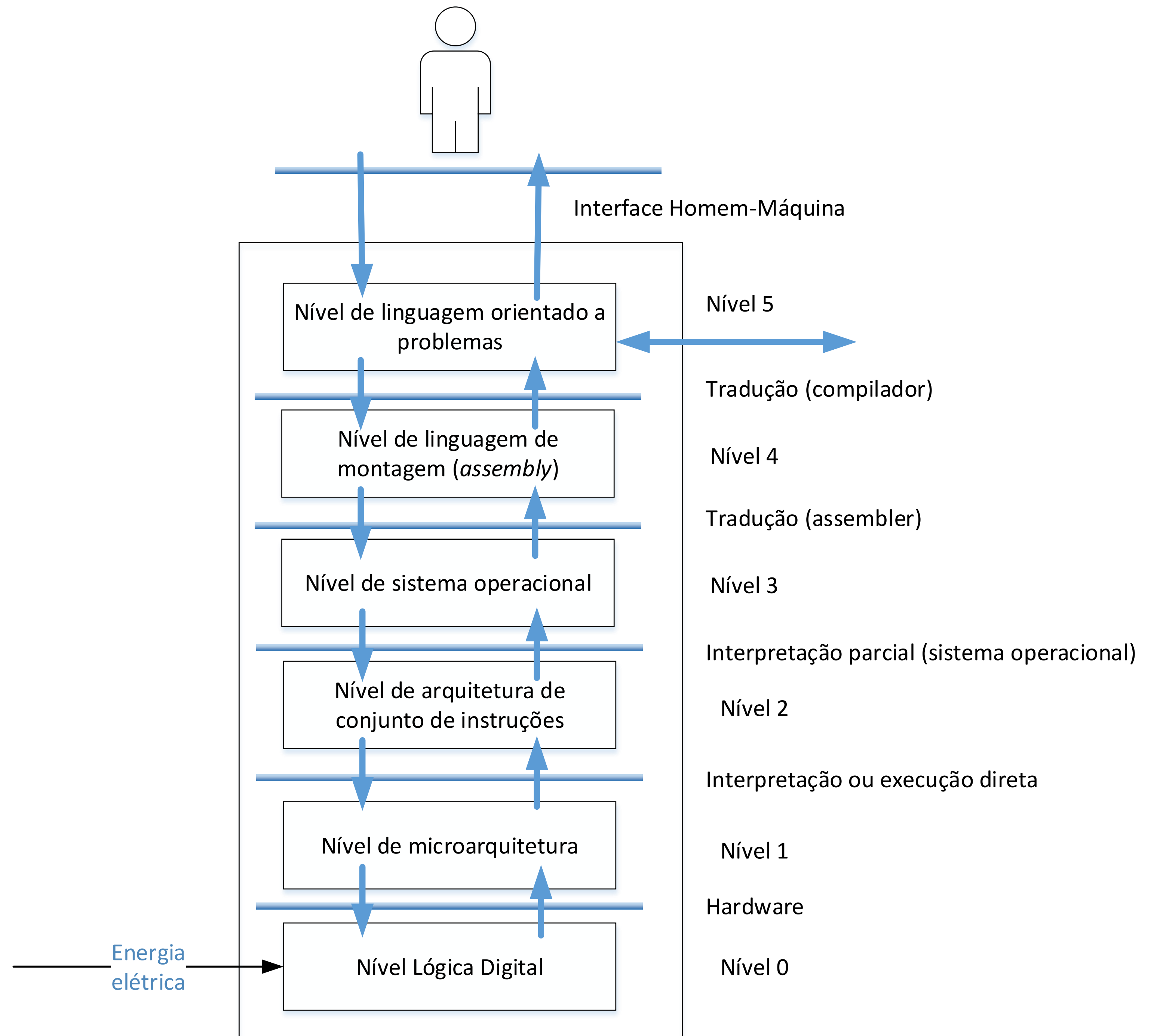
Arquitetura de Computadores

Prof. Clayton J A Silva, MSc

clayton.silva@professores.ibmec.edu.br




Máquina de seis níveis





Sistemas operacionais:
conceitos e mecanismos
Prof. Carlos Maziero



O que é um sistema operacional

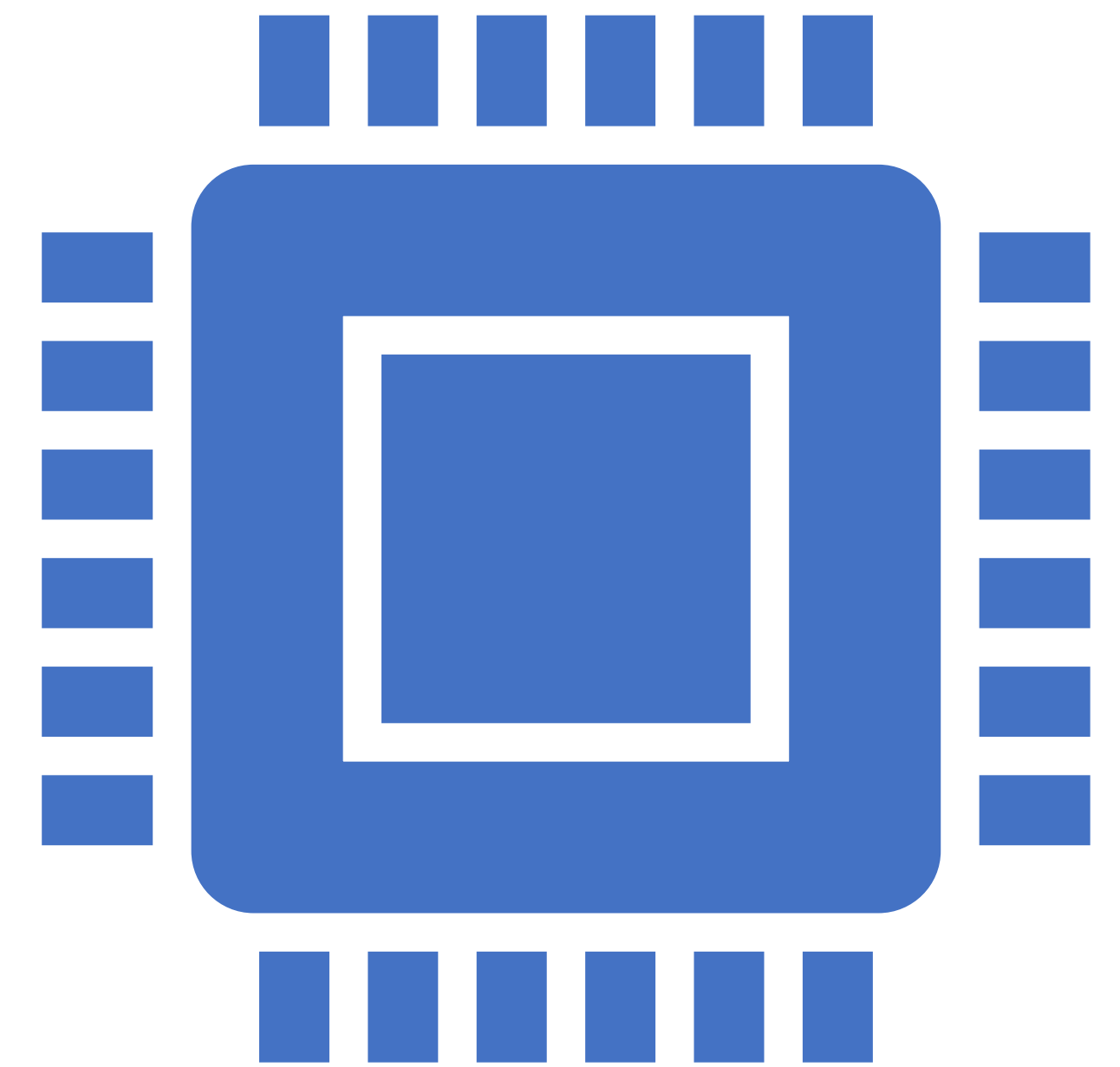
- O conjunto de instruções do nível do S.O. é o conjunto completo de instruções disponíveis para os programadores de aplicação. **Contém todas as instruções de nível ISA**, bem como o conjunto de **novas instruções** que o sistema operacional adiciona.
- Essas novas instruções são denominadas **chamadas de sistema**.
- O nível do sistema operacional é sempre **interpretado**.

O que é um sistema operacional

- Contém todas as instruções de nível ISA, bem como o conjunto de novas instruções
- **Chamadas de sistema**
- O nível do sistema operacional é sempre **interpretado**

Chamadas de sistema

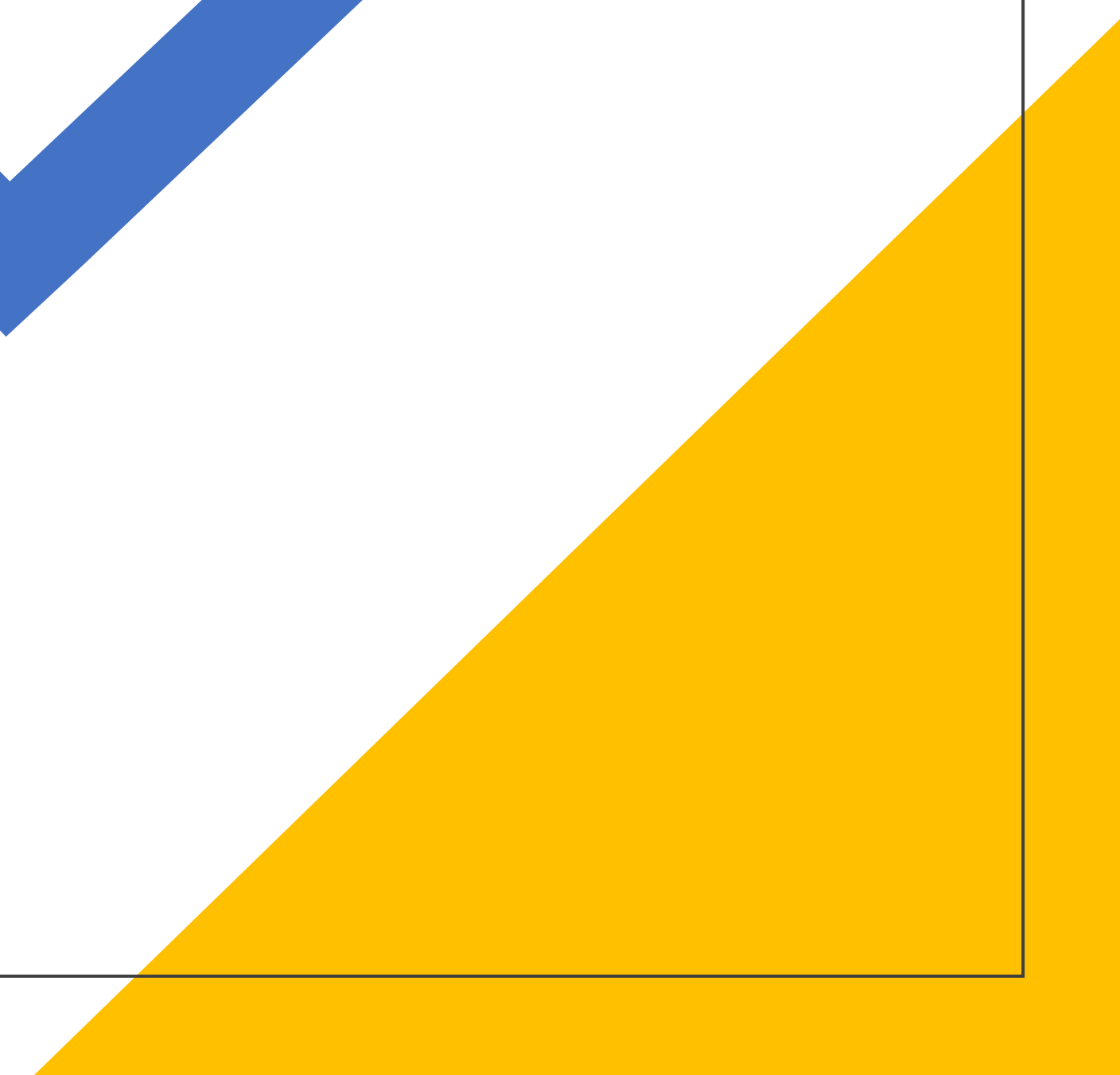
- Mecanismo de interrupção utilizado pelas aplicações para serem atendidos pelo sistema
- Geralmente é oferecida as aplicações pela **biblioteca do sistema (*system library*)**: prepara os parâmetros, invoca a chamada e devolve os resultados
- O conjunto de chamadas de sistema define a **API (*Application Programming Interface*)**



Objetivos

1. Conveniência. Visa tornar o uso do computador mais conveniente.
2. Eficiência. Permite uma utilização mais eficiente dos recursos do sistema.

Atividades típicas



Atividades típicas do SO

- a. **Criação de programas.** Auxiliar o programador no desenvolvimento de programas. Fornece programas utilitários, os quais na verdade não são parte do sistema operacional, mas podem ser acessados por ele.
- b. **Execução de programas.** Executa várias tarefas na execução de programas, como carregar instruções e dados na memória, inicialização de arquivos e dispositivos de entrada e saída, entre outras.

Atividades típicas do SO

c. **Acesso aos dispositivos de entrada e saída.** Encarrega-se do gerenciamento desses dispositivos, que possui um conjunto peculiar de instruções.

d. **Acesso controlado aos arquivos.** Controla o formato dos arquivos no meio de armazenamento.

Atividades típicas do SO

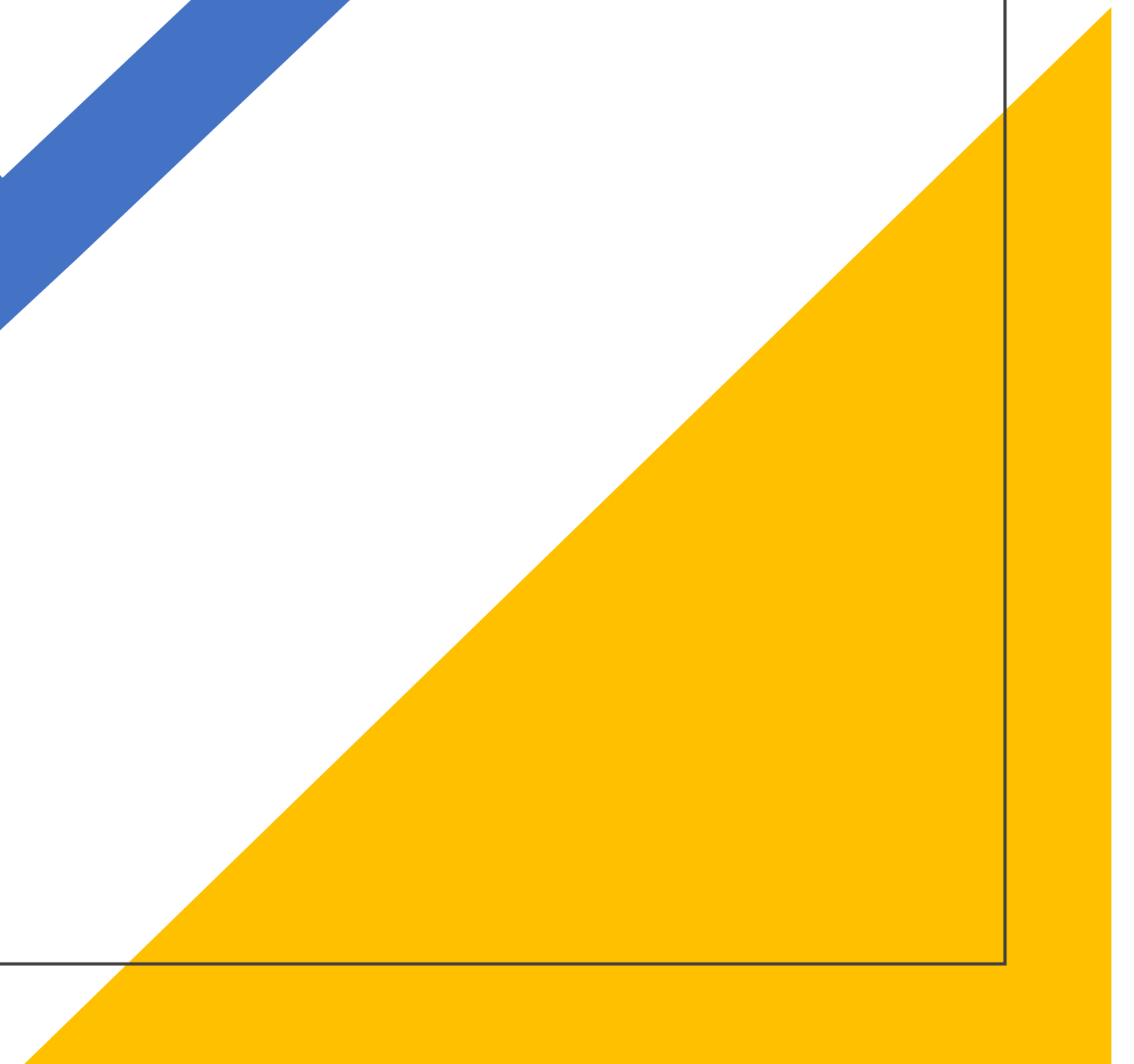
e. **Acesso ao sistema.** No caso de sistemas compartilhados ou públicos, controla o acesso ao sistema como um todo e acesso a recursos específicos. Fornece proteção contra uso não autorizado tanto para recursos quanto para dados de usuários. Além disso, resolve conflitos em caso de contenção de um recurso.

f. **Deteccção e reação aos erros.** Reage e elimina os erros do sistema, tanto de software quanto de hardware.

g. **Monitoramento.** Monitora parâmetros de desempenho.

Atividades típicas

- a. Criação de programas
- b. Execução de programas
- c. Acesso aos dispositivos de E/S
- d. Acesso controlado aos arquivos
- e. Acesso ao sistema
- f. Detecção e reação aos erros
- g. Monitoramento



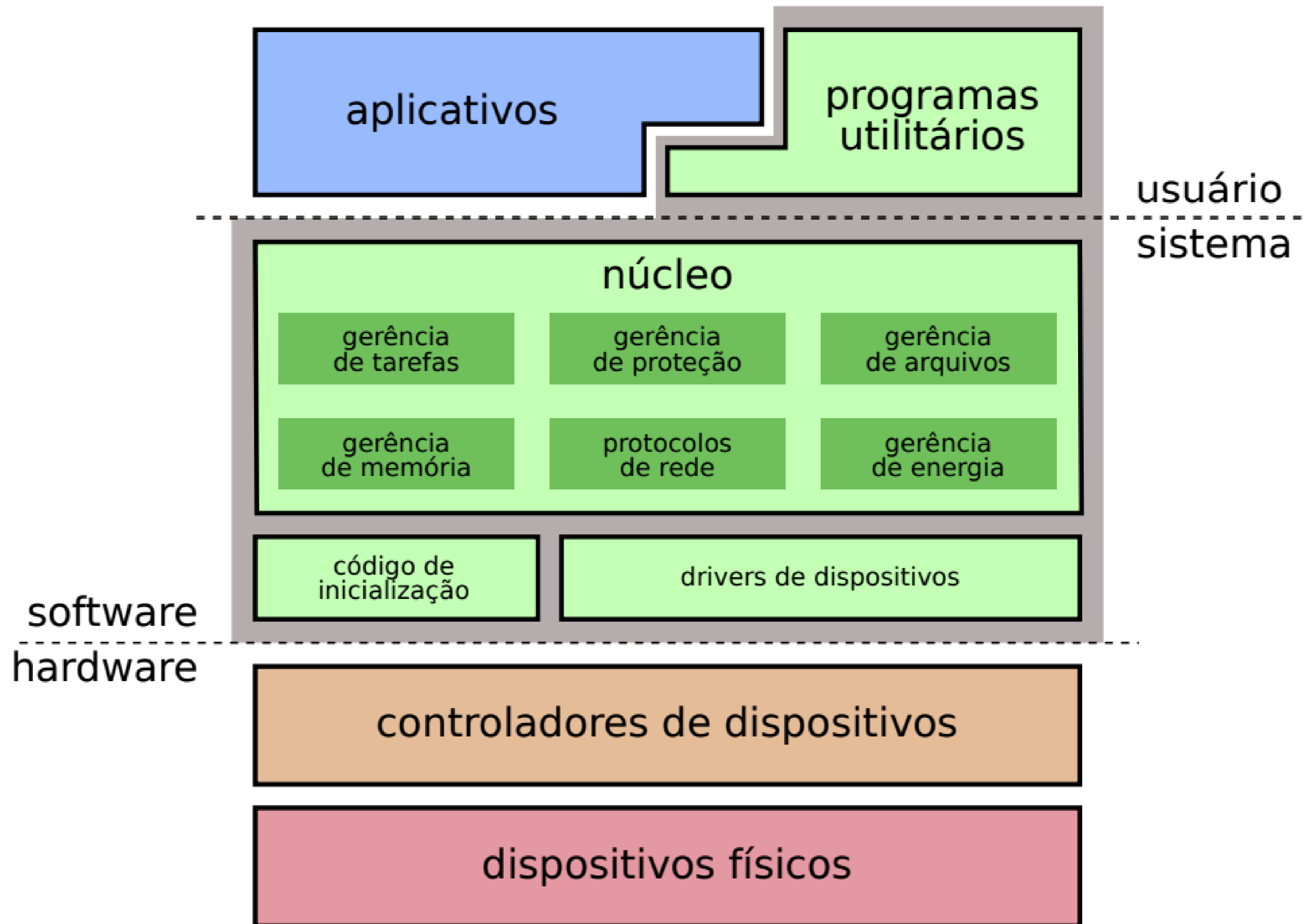
Elementos do sistema operacional

Núcleo (*kernel*)

Código de inicialização (*boot code*)

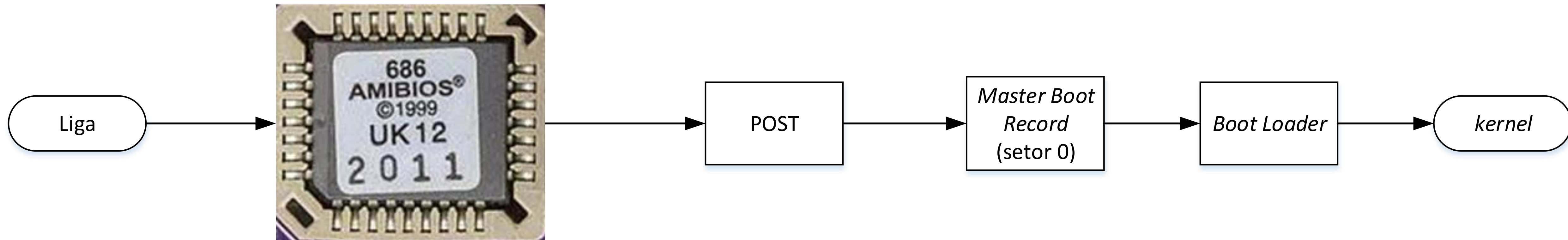
Drivers

Programas utilitários



Código de inicialização

- Reconhecer os dispositivos instalados, testá-los e configurá-los adequadamente para seu uso posterior.
- Carregar o núcleo do sistema operacional em memória e iniciar sua execução.



System Time	[14:28:46]
System Date	[Fri 07/24/2009]
Legacy Diskette A	[1.44M, 3.5 in.]
Language	[English]
▶ Primary IDE Master	: [Not Detected]
▶ Primary IDE Slave	: [PHILIPS DVD+/-RW D]
▶ Third IDE Master	: [SAMSUNG HD300LJ]
▶ Third IDE Slave	: [SAMSUNG SP2504C]
▶ Fourth IDE Master	: [Not Detected]
▶ Fourth IDE Slave	: [Not Detected]
▶ IDE Configuration	

While entering setup, BIOS auto detects the presence of IDE devices. This displays the status of auto detection of IDE devices.

↔ Select Screen
↑↓ Select Item

▶ System Information

Interface de configuração da BIOS

Go to Sub Screen

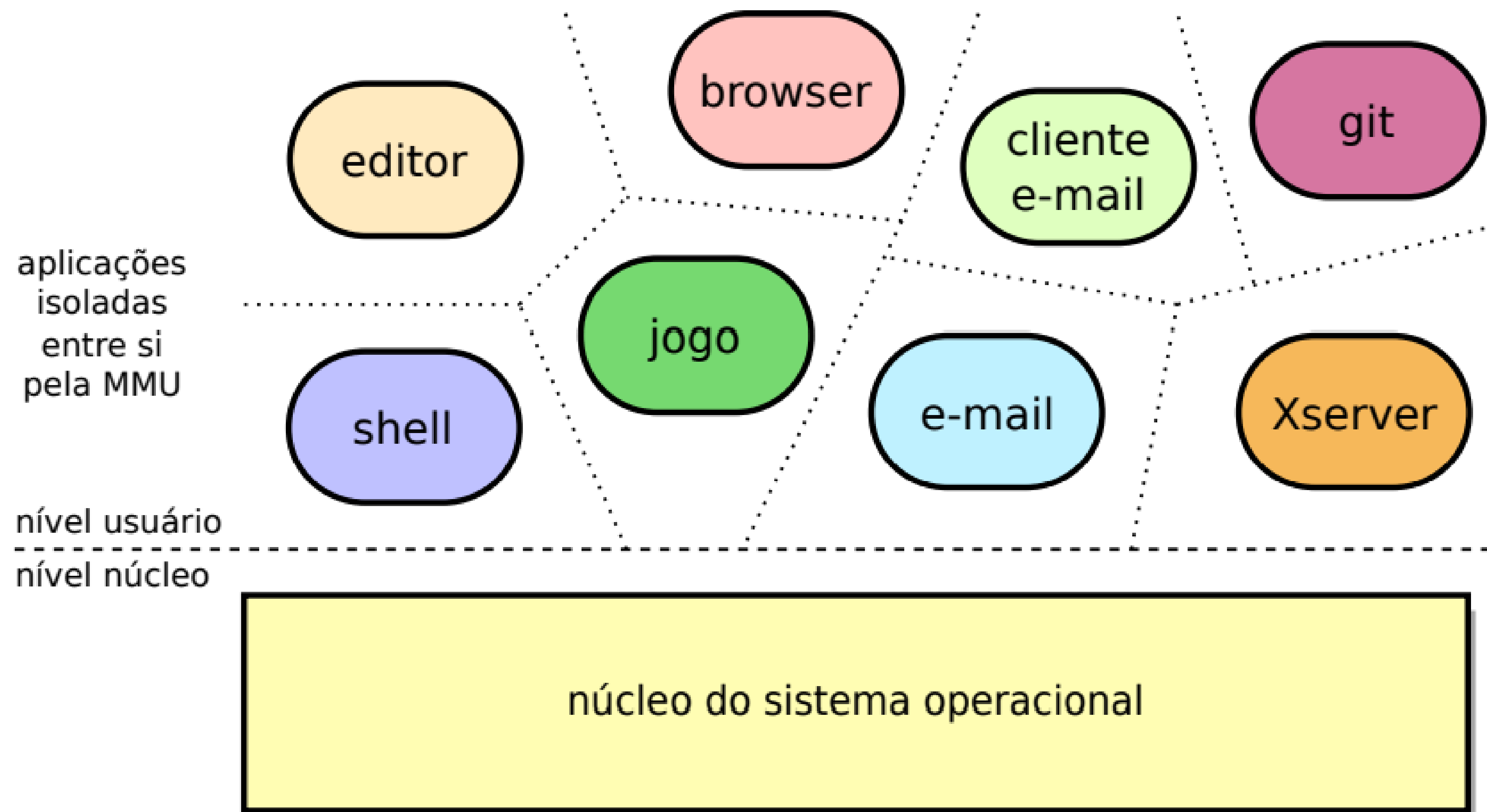
F1 General Help

F10 Save and Exit

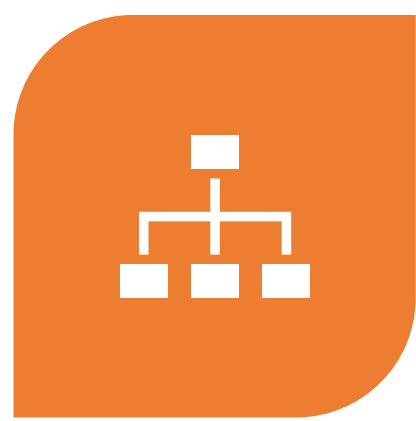
ESC Exit

Níveis de privilégio

- Núcleo
- Usuário



Áreas do sistema operacional



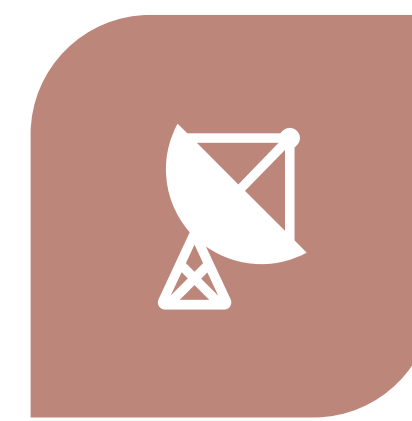
GESTÃO DE PROCESSOS:
CRIAR, CARREGAR
CÓDIGO,
TERMINAR,
ESPERAR,
LER/MUDAR
ATRIBUTOS.



GESTÃO DA MEMÓRIA:
ALOCAR/LIBERAR/
MODIFICAR ÁREAS
DE MEMÓRIA.



GESTÃO DE ARQUIVOS: CRIAR,
REMOVER, ABRIR,
FECHAR, LER,
ESCREVER,
LER/MUDAR
ATRIBUTOS.



COMUNICAÇÃO:
CRIAR/DESTRUIR
CANAIS DE
COMUNICAÇÃO,
RECEBER/ENVIAR
DADOS.

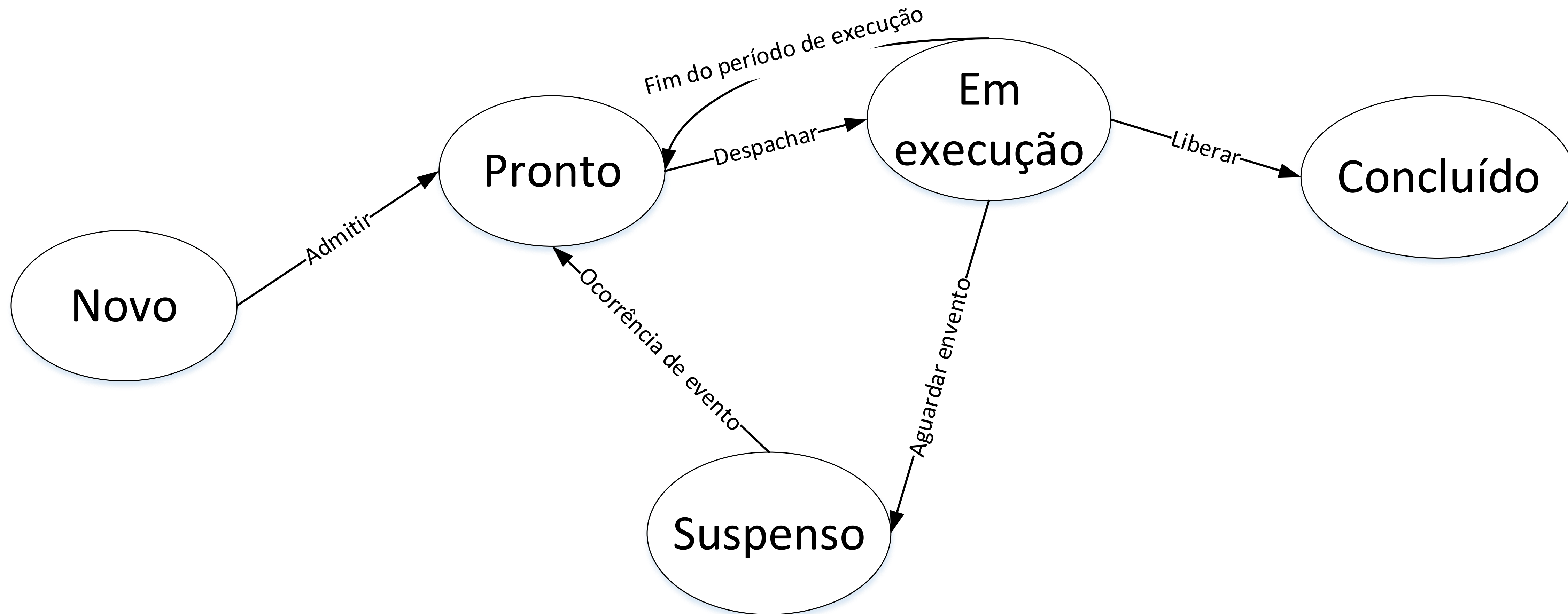


GESTÃO DE DISPOSITIVOS:
LER/MUDAR
CONFIGURAÇÕES,
LER/ESCREVER
DADOS.

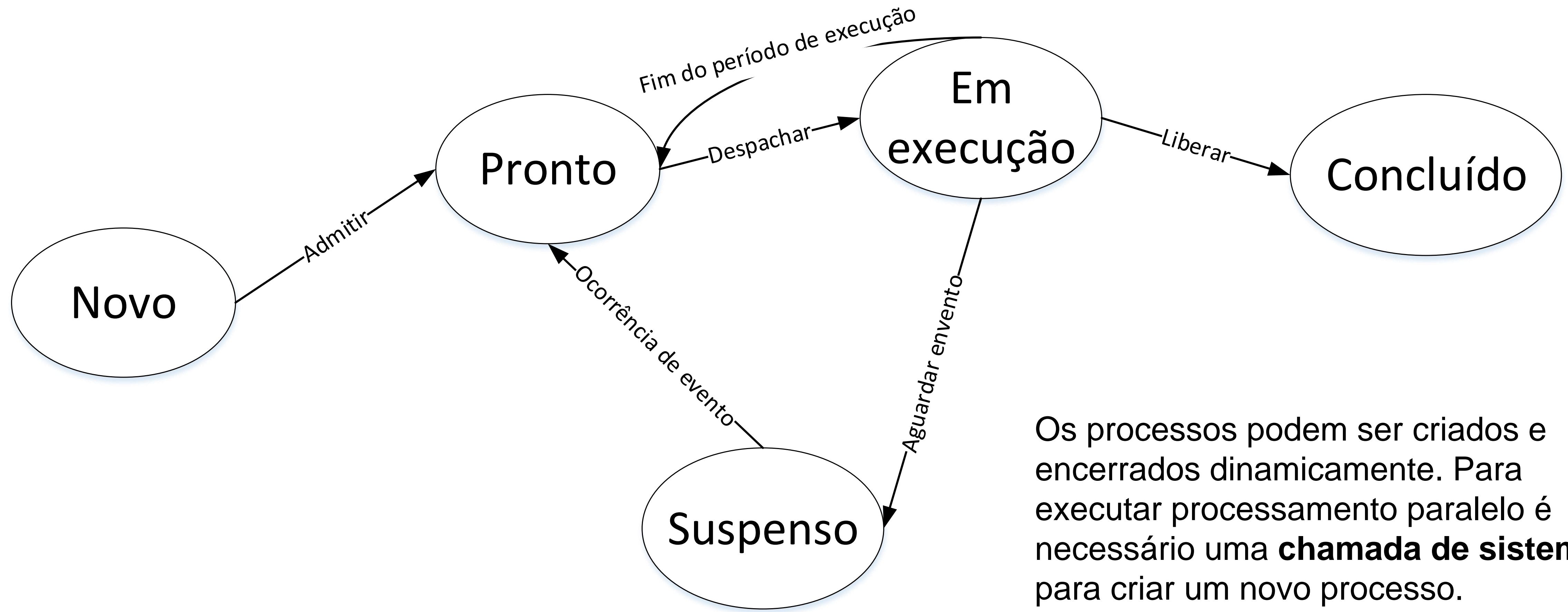


GESTÃO DO SISTEMA:
LER/MUDAR DATA E
HORA,
DESLIGAR/SUSPENDER/
REINICIAR O SISTEMA.

Processos



Modelo de processo com cinco estados



Bloco de controle de processos do sistema operacional

Informação	Descrição
Identificador	Cada processo tem um identificador distinto.
Estado	Um dos cinco estados possíveis
Contexto	Dados contidos nos registradores do processador. Dados do Contador de Programas do processador. Os dados são guardados pelo Sistema Operacional quando a execução do processo é suspensa.
Estado de E/S	Dados sobre os requisitos pendentes de E/S. Dispositivos alocados. Lista de arquivos alocados.
Contabilidade	Quantidade de tempo do processador. Tempo total já usado pelo processo. Limites de tempo de execução. Contabilização de uso dos recursos.

Escalonamento de processos

O escalonamento é a decisão do sistema operacional de acrescentar ou executar processos já disponíveis na memória.

Escalonamento de longo prazo

Escalonamento de médio prazo

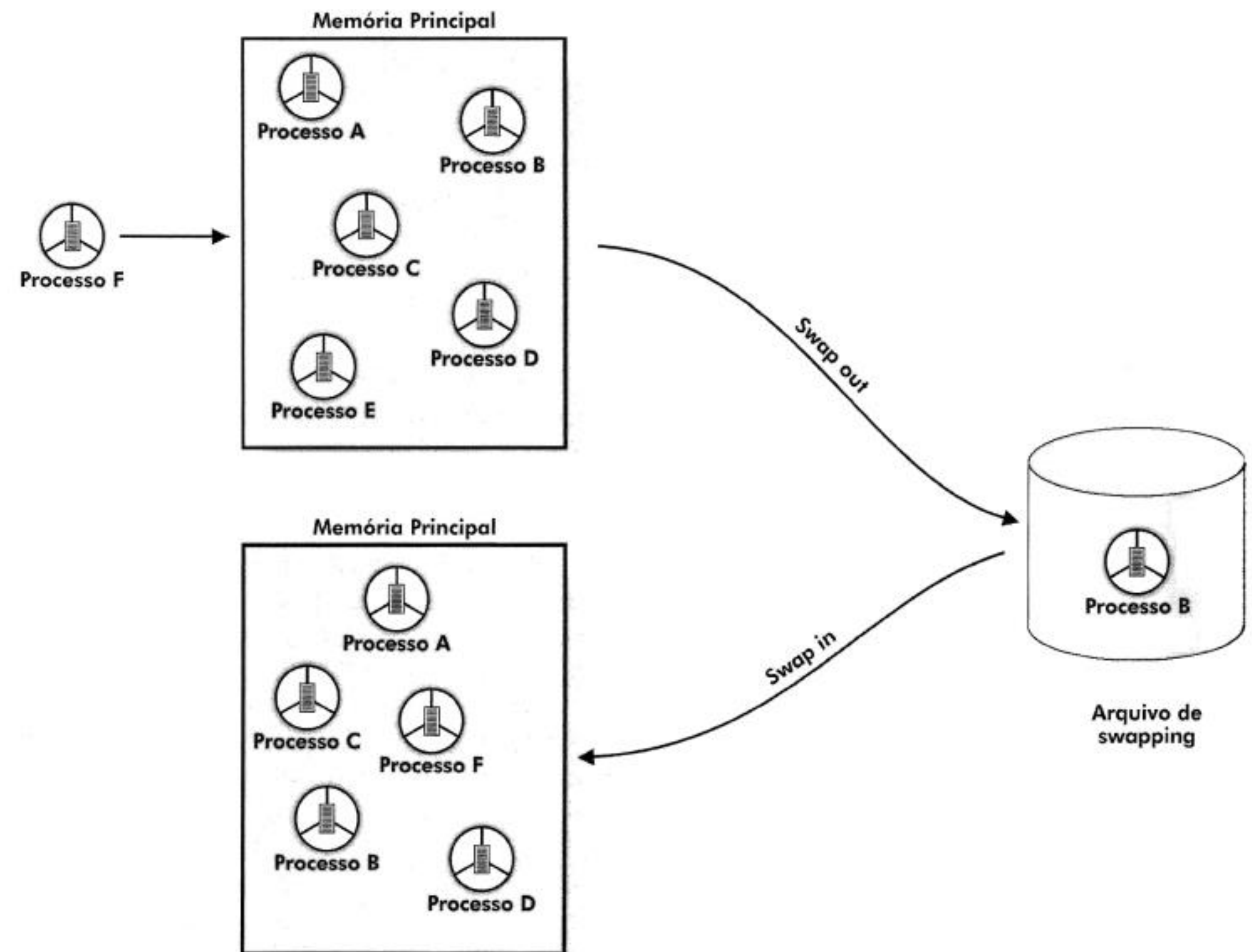
Escalonamento de curto prazo

Escalonamento de processos

Pode ser a **longo prazo**, quando a decisão é a de **acrescentar um novo processo** ao conjunto de processos a serem executados pelo processador. O sistema operacional deve **inicializar o processo** para colocá-lo no estado de pronto.

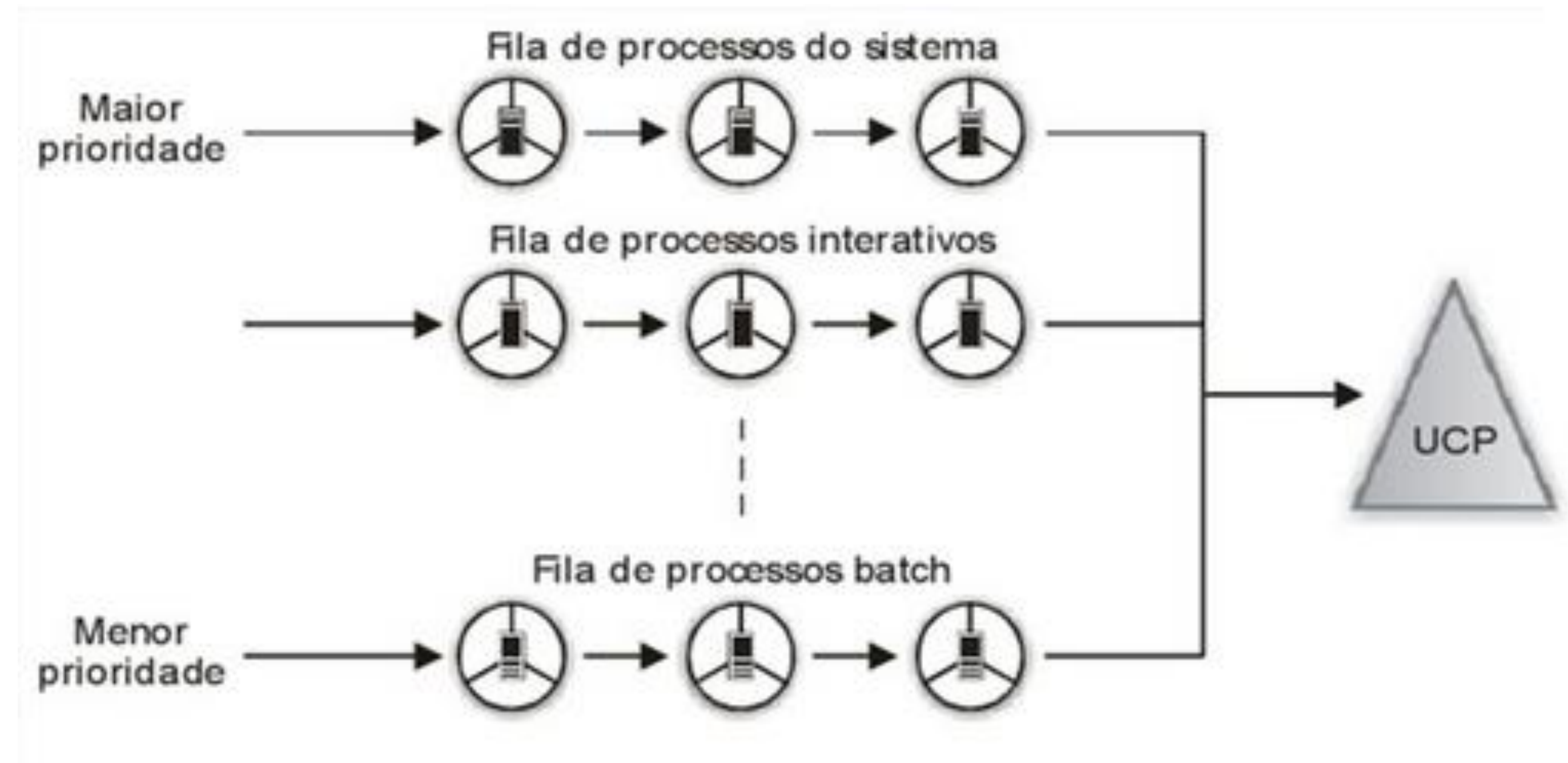
Escalonamento de processos

- O escalonamento também pode ser a **médio prazo**, quando a decisão é de acrescentar um processo a um conjunto de processos que estão parcial ou completamente carregados na memória principal. Faz parte da função de **troca de processos (*swapping*)** entre a memória principal e a memória secundária



Escalonamento de processos

- No **escalonamento de curto prazo** o S.O. define qual dos processos disponíveis na memória principal será executado pelo processador.
- Decisão é realizada frequentemente pelo despachante (*dispatcher*).
- A operação do escalonador requer a definição do estado do processo.



Sistema Operacional

Chamadas ao sistema feitas
por um processo

Rotina de tratamento
de chamadas ao
sistema

Interrupção do processo

Interrupção de E/S

Rotina de tratamento
de interrupções

Fila de longo
prazo

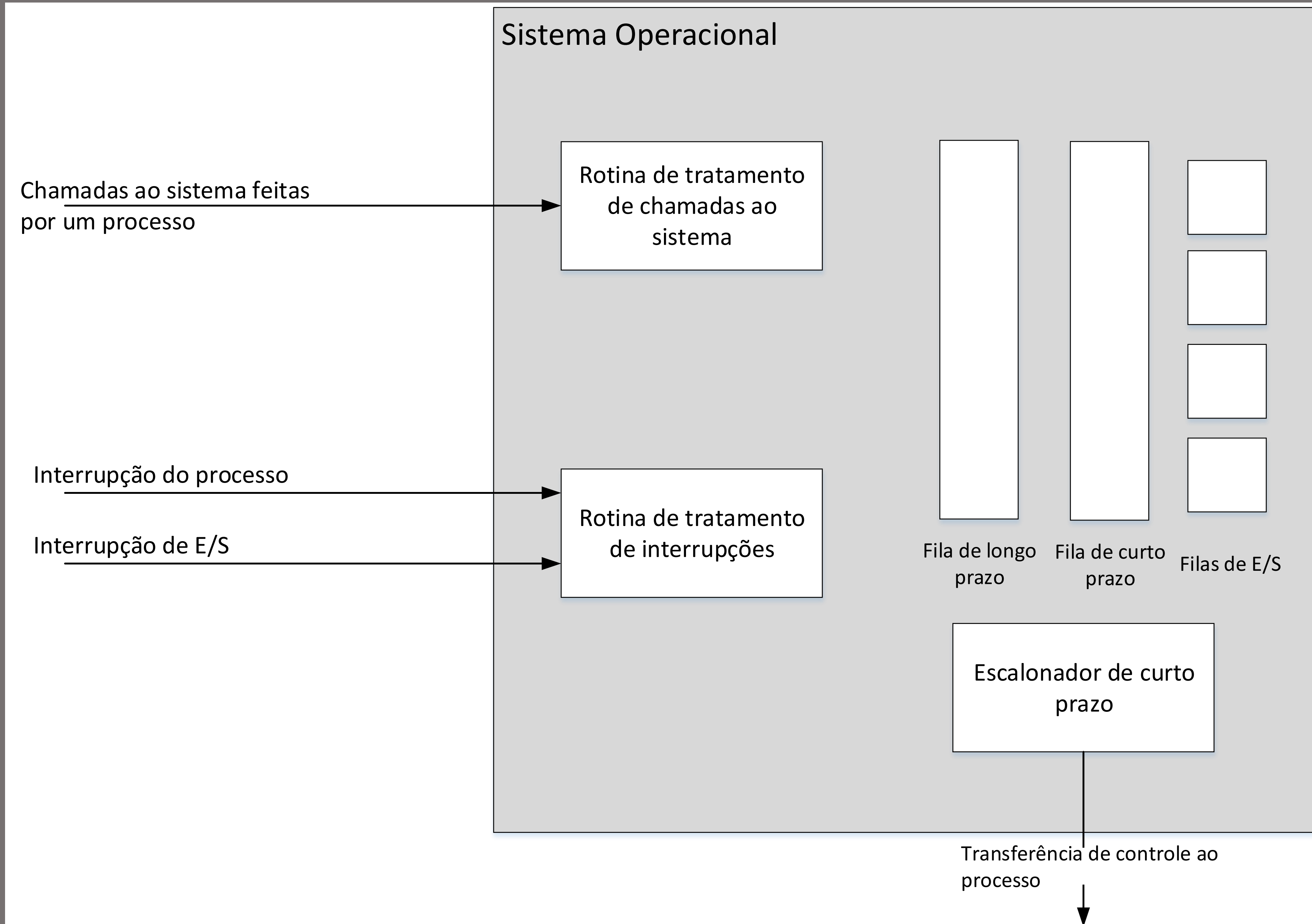
Fila de curto
prazo

Filas de E/S

Escalonador de curto
prazo

Transferência de controle ao
processo

Escalonamento
e filas de
processos



Tarefas x processos

Processo para sistemas operacionais foi introduzido de forma mais genérica do que tarefa (*job*), mas atualmente pode ser definido como uma **entidade para a qual o processador é alocado, como um programa.**

Tarefas x processos

- Processo: unidade de contexto, ou seja, contêiner de recursos utilizados por uma ou mais tarefas para sua execução
- Um processo pode então conter várias tarefas, que compartilham esses recursos.

Gerência de memória

Memória virtual

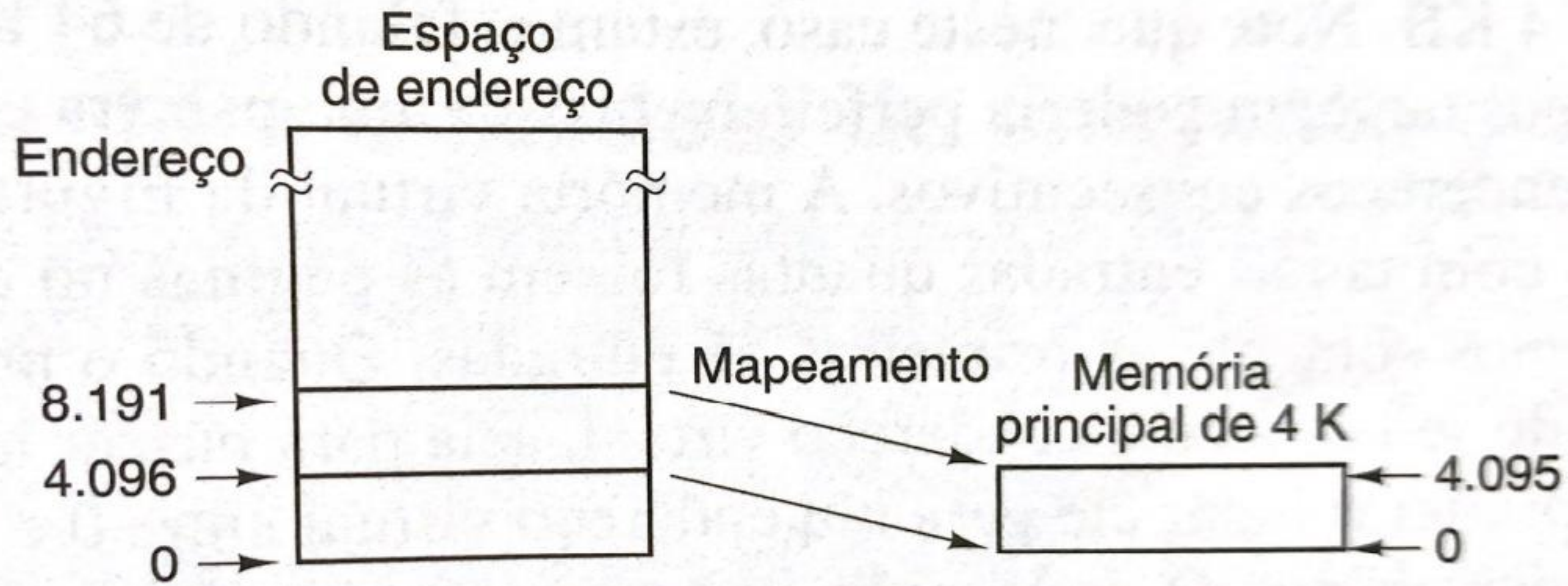
Paginação

Segmentação

Gerência de memória

- **Memória virtual** – Uso de disco como memória lógica - virtual
- **Paginação.** Os trechos de programa lidos em memória secundária são organizados em **páginas**, todas do **mesmo tamanho**.
- O espaço de endereços contempla uma parte que endereça a memória física e outra parte endereça a memória virtual. O **espaço de endereço físico** e o **espaço de endereço virtual** são bem definidos.

Mapeamento endereços virtuais para endereços da memória principal



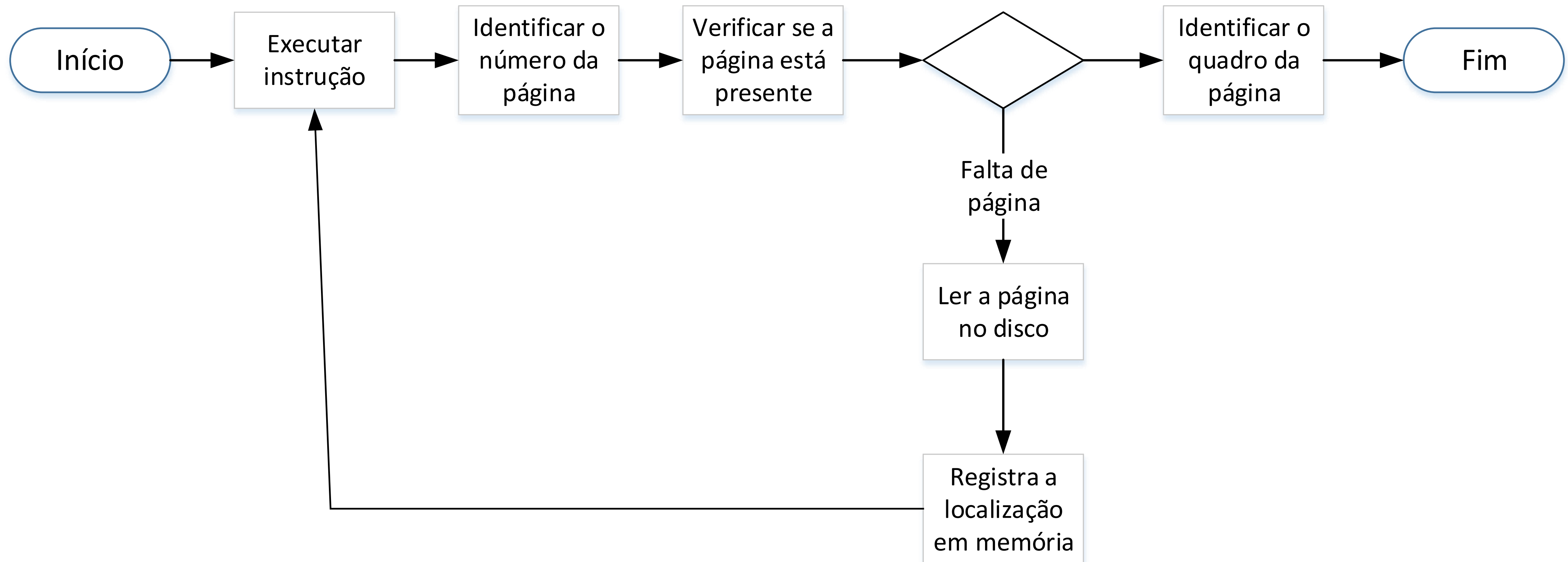
Paginação

- Um mapa de memória ou tabela de páginas especifica o endereço físico correspondente a cada endereço virtual.
- O espaço de endereço físico é dividido em quadros de página
- As máquinas que utilizam memória virtual dispõem de um dispositivo chamado MMU – Unidade de Gerenciamento de Memória

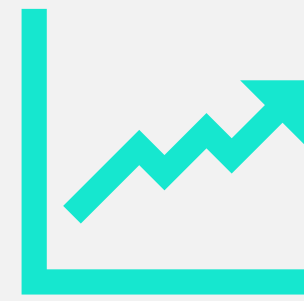
Página	Endereços virtuais
15	61.440 – 65.535
14	57.344 – 61.439
13	53.248 – 57.343
12	49.152 – 53.247
11	45.056 – 49.151
10	40.960 – 45.055
9	36.864 – 40.959
8	32.768 – 36.863
7	28.672 – 32.767
6	24.576 – 28.671
5	20.480 – 24.575
4	16.384 – 20.479
3	12.288 – 16.383
2	8.192 – 12.287
1	4.096 – 8.191
0	0 – 4.095

32 K da parte inferior da memória principal	
Quadro de página	Endereços físicos
7	28.672 – 32.767
6	24.576 – 28.671
5	20.480 – 24.575
4	16.384 – 20.479
3	12.288 – 16.383
2	8.192 – 12.287
1	4.096 – 8.191
0	0 – 4.095

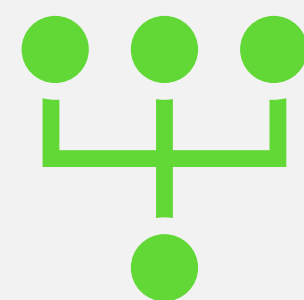
Paginação



No início, o carregamento das páginas é por demanda...



Paginação por demanda



Paginação utilizando o conjunto de trabalho – princípio da localidade

- LRU - *Least Recently Used*
- FIFO – *First-In-First-Out*

Substituindo
as páginas
da
memória...

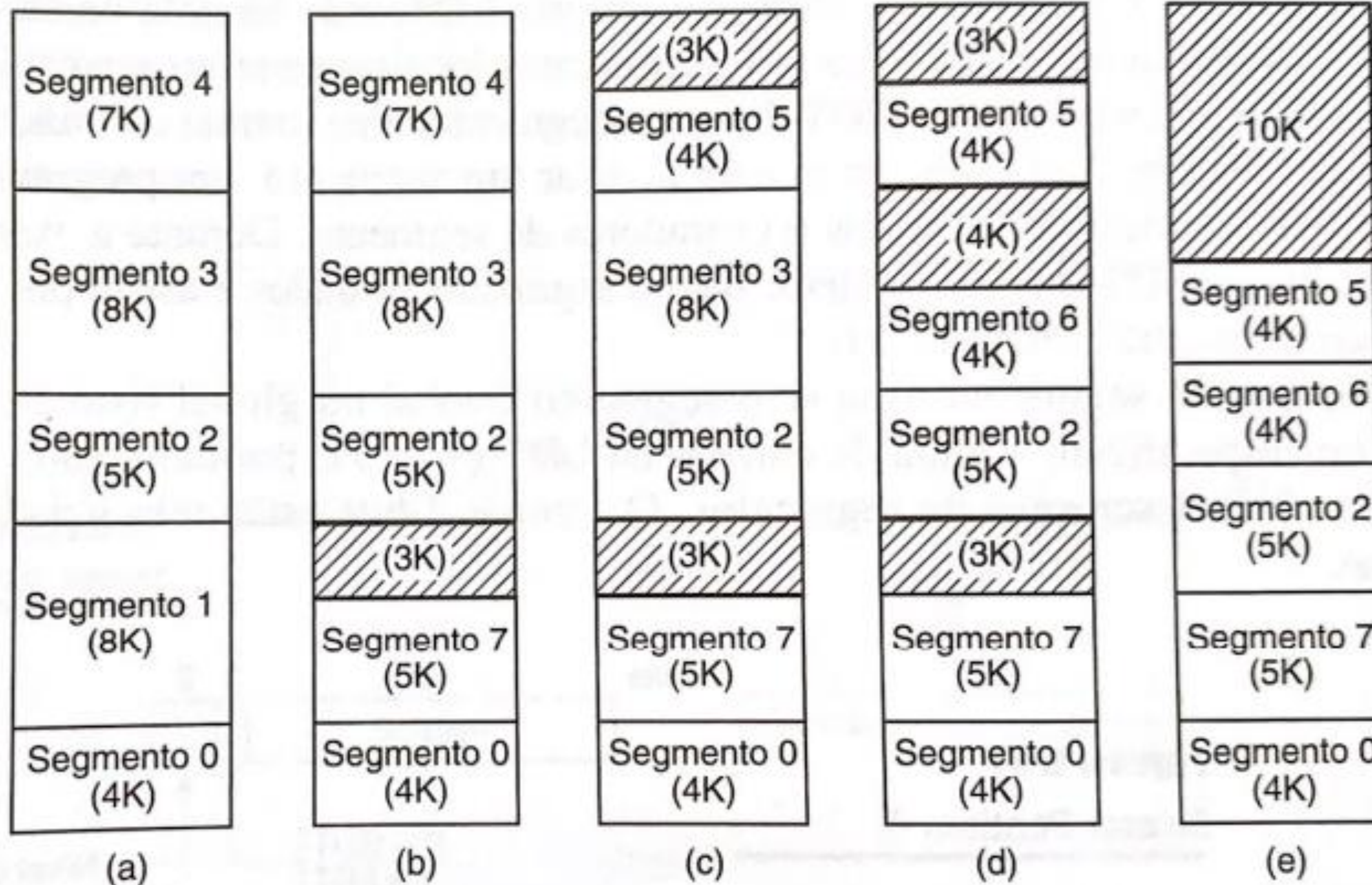
Segmentação

- Uma solução aos problemas advindos de ter somente um espaço de memória virtual pode ser simplesmente **prever muitos espaços de endereço virtual completamente independentes**, denominados **segmentos**.
- O segmento é uma **entidade lógica**, da qual o programador está ciente e que a usa conscientemente. O segmento pode conter um procedimento, um vetor, uma pilha, enfim vários tipos de estrutura de dados.

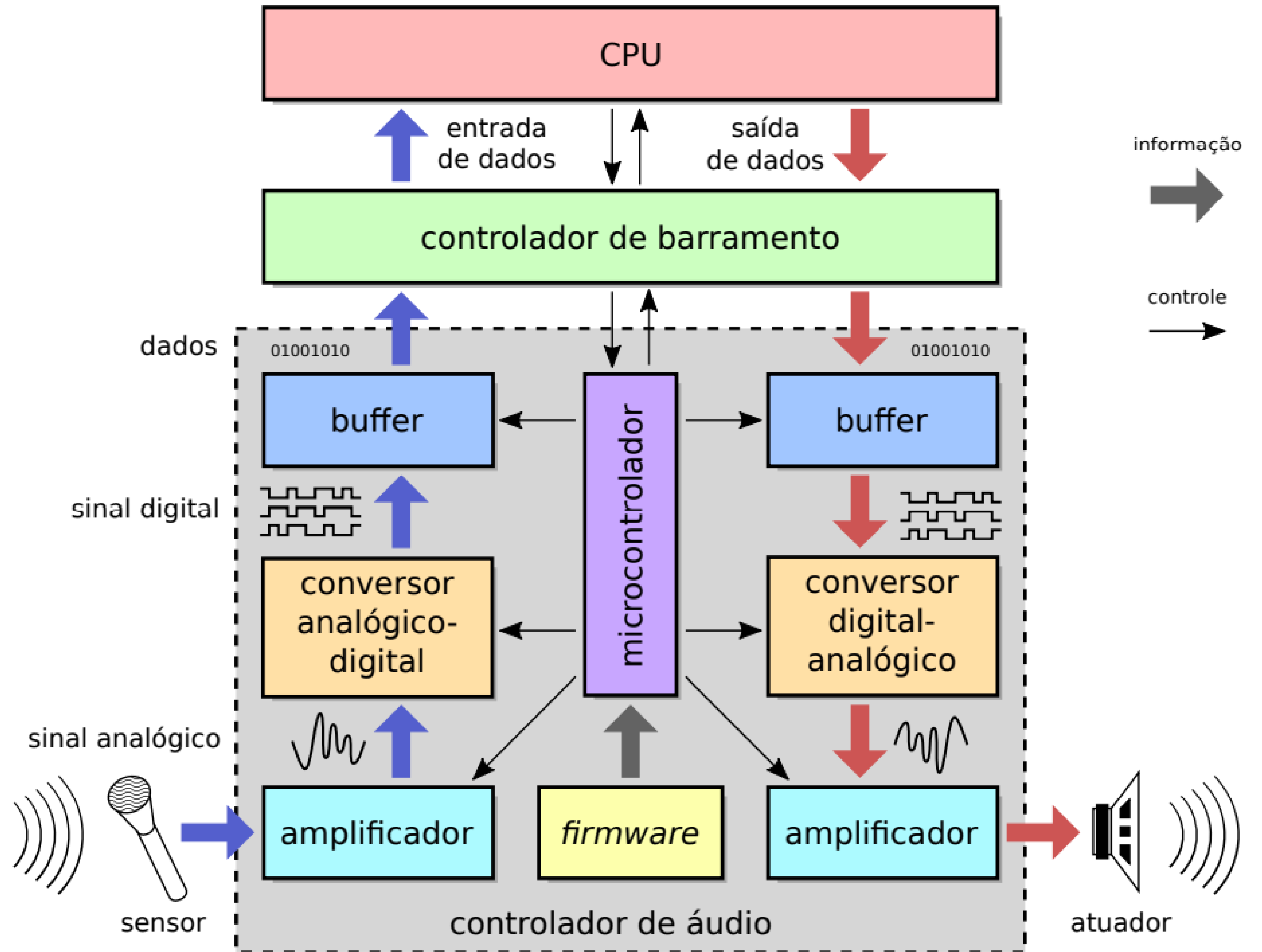
Formas de implementar a segmentação...

- **Permutação** - Se for feita referência a um segmento que não está na memória naquele instante, então o segmento é trazido à memória. Se não houver mais espaço para ele, um ou mais segmentos devem ser escritos na memória secundária antes.
- Paginação

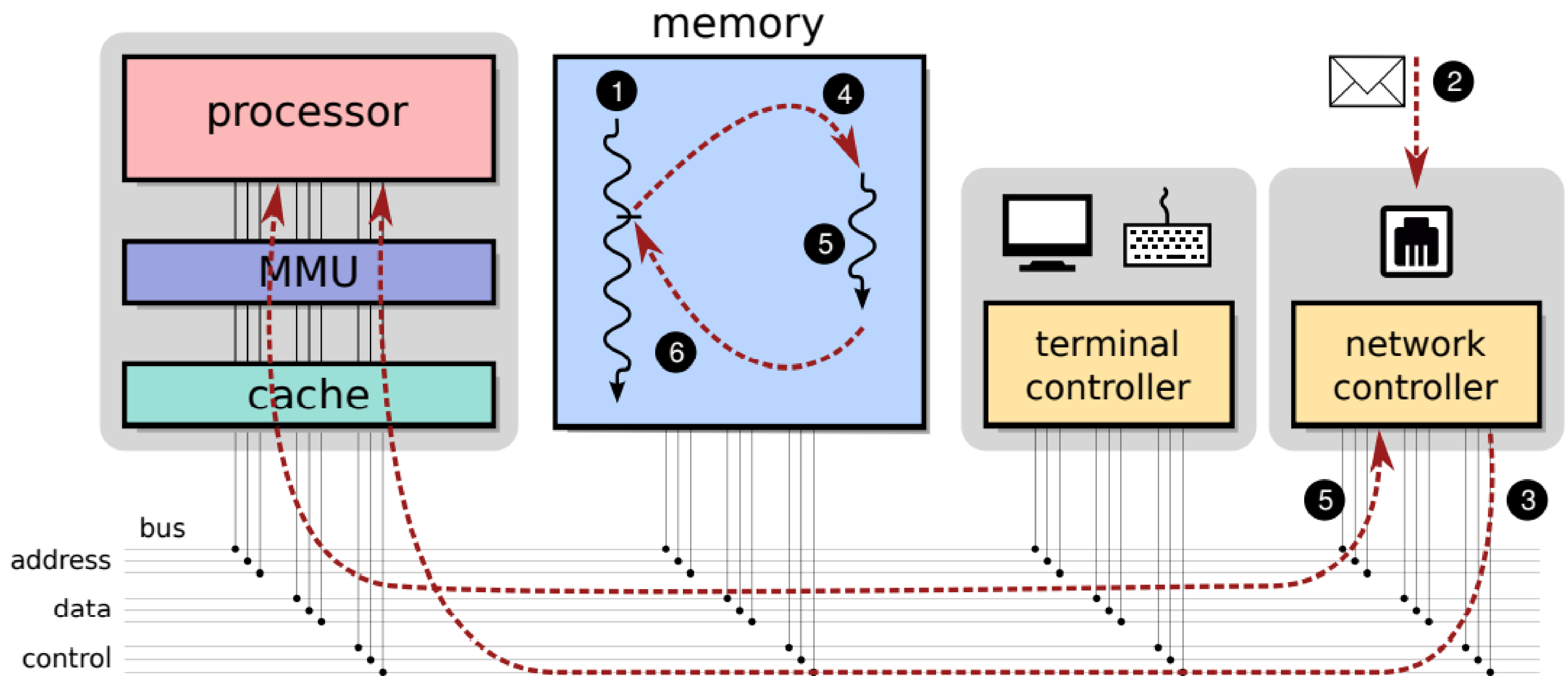
Fragmentação externa



Gestão de entrada/saída



Interrupções



Gestão de entrada/saída

- Comunicação controlador-processador
- Requisição de interrupção (*Interrupt Request- IRq*)
- As requisições de interrupção são sinais elétricos veiculados através do barramento de controle do computador.
- Cada interrupção está geralmente associada a um número inteiro que permite identificar sua origem.

Dispositivo	Interrupção
teclado	1
interface serial COM2	3
interface serial COM1	4
interface paralela LPT1	7
relógio de tempo real	8
mouse PS/2	12
barramento ATA primário	14
barramento ATA secundário	15

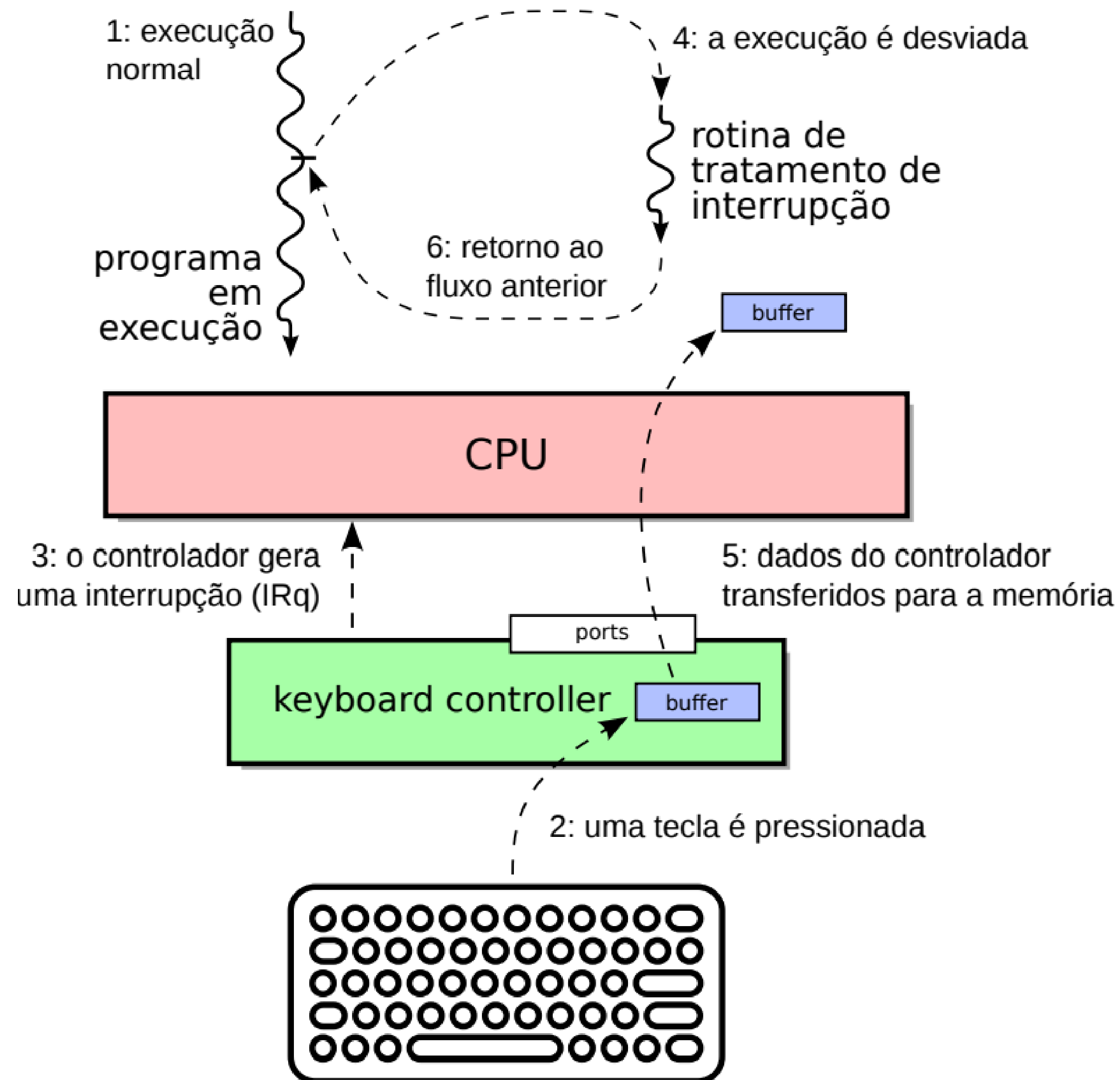
Tratamento de interrupções

- Cada requisição de interrupção deve disparar uma rotina de tratamento específica.
- Tabela de endereços de funções: Tabela de Interrupções (*IVT - Interrupt Vector Table*)

onde cada entrada da tabela contém o endereço da rotina de tratamento da interrupção correspondente.

- as interrupções geradas pelos dispositivos de entrada/saída não são transmitidas diretamente ao processador, mas a um controlador de interrupções programável (*PIC - Programmable Interrupt Controller*)

Roteiro típico de processamento de interrupção



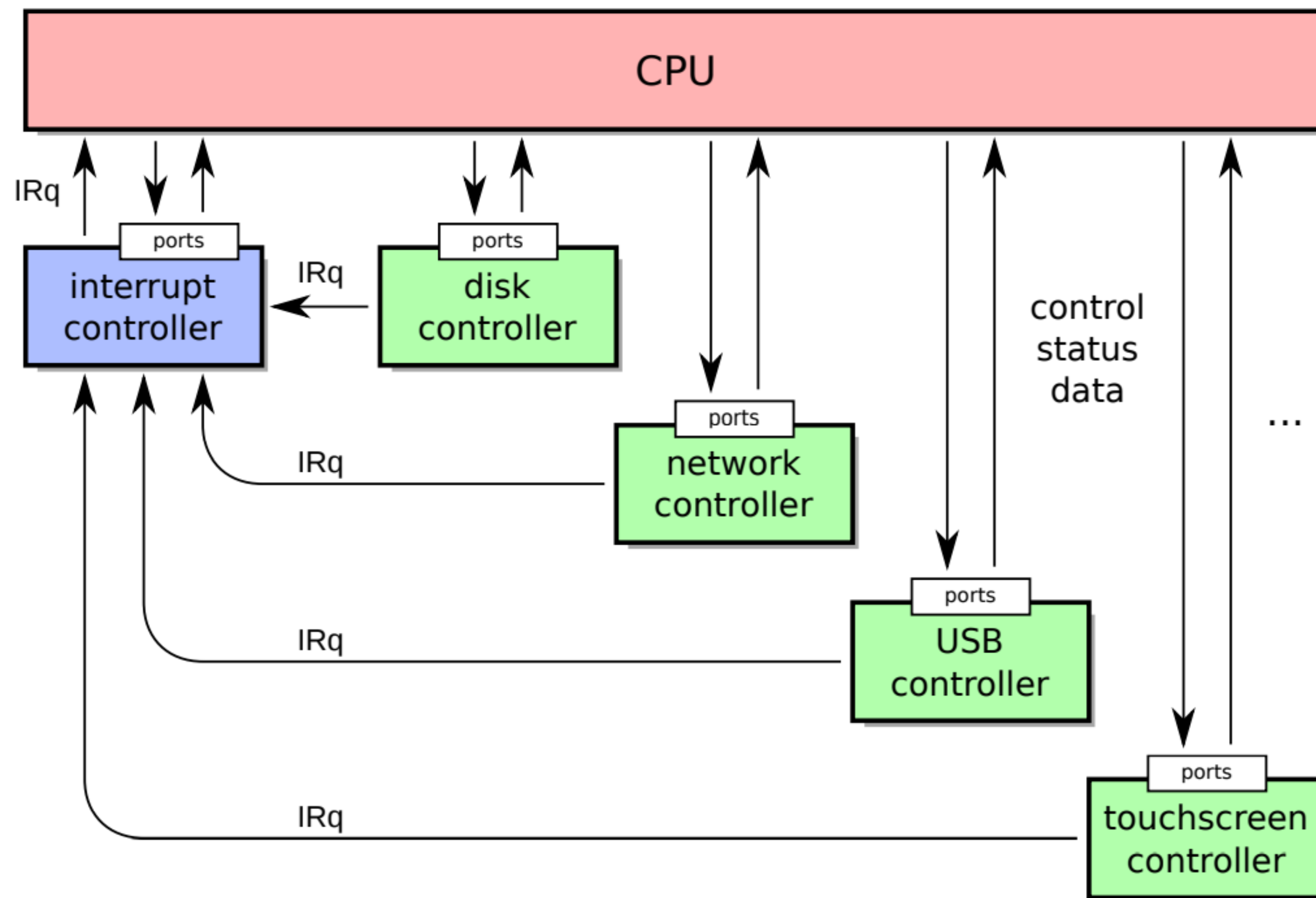
Drivers

- Módulos de código específicos para acessar os dispositivos físicos.
- Existe um driver para cada tipo de dispositivo
- Muitas vezes o driver é construído pelo próprio fabricante do hardware e fornecido em forma compilada (em linguagem de máquina) para ser acoplado ao restante do sistema operacional.

Exceções

- Eventos internos ao processador que ocasionam o desvio da execução usando o mesmo mecanismo de interrupção

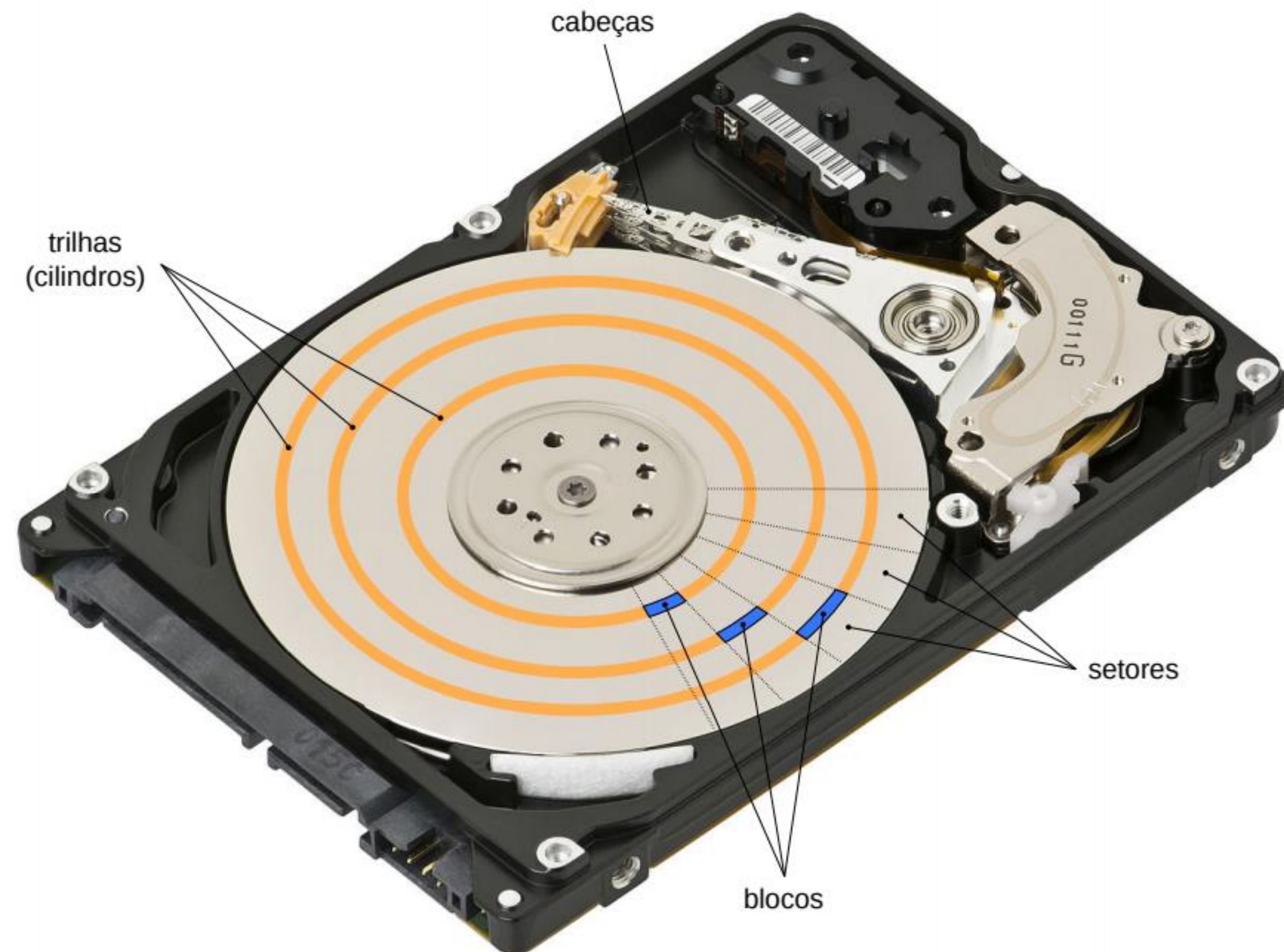
Exceção	Descrição
0	erro de divisão por zero
3	breakpoint (parada de depurador)
5	erro de faixa de valores
6	operação inválida
7	dispositivo não disponível
11	segmento de memória ausente
12	erro de pilha
14	falta de página
16	erro de ponto flutuante
19-31	valores reservados pela Intel



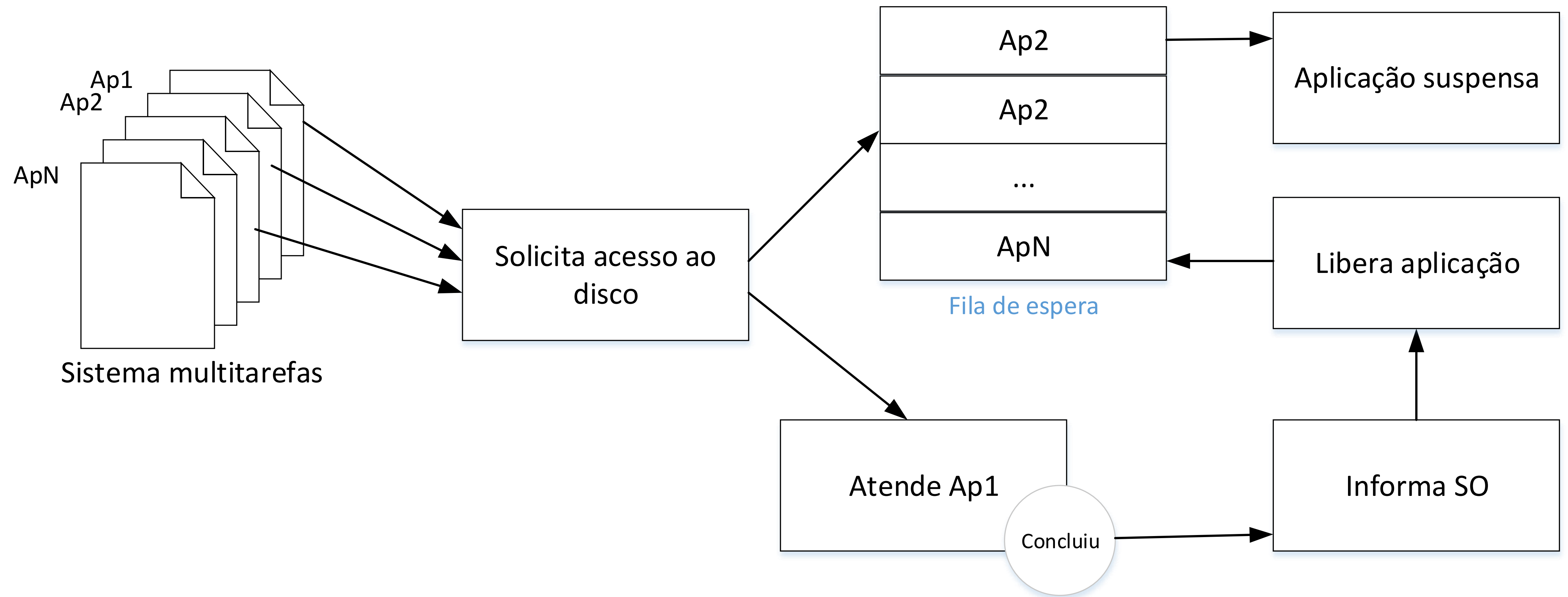
Tratamento
de
interrupções

Dados no disco rígido

- Cada **face** é dividida logicamente em
 - **trilhas** (ou cilindros) e
 - **setores**.
- A interseção de trilha e setor em uma face define um **bloco físico**, que é a unidade básica de armazenamento e transferência de dados no disco.
- Atualmente os blocos possuem **4.096 bytes**.

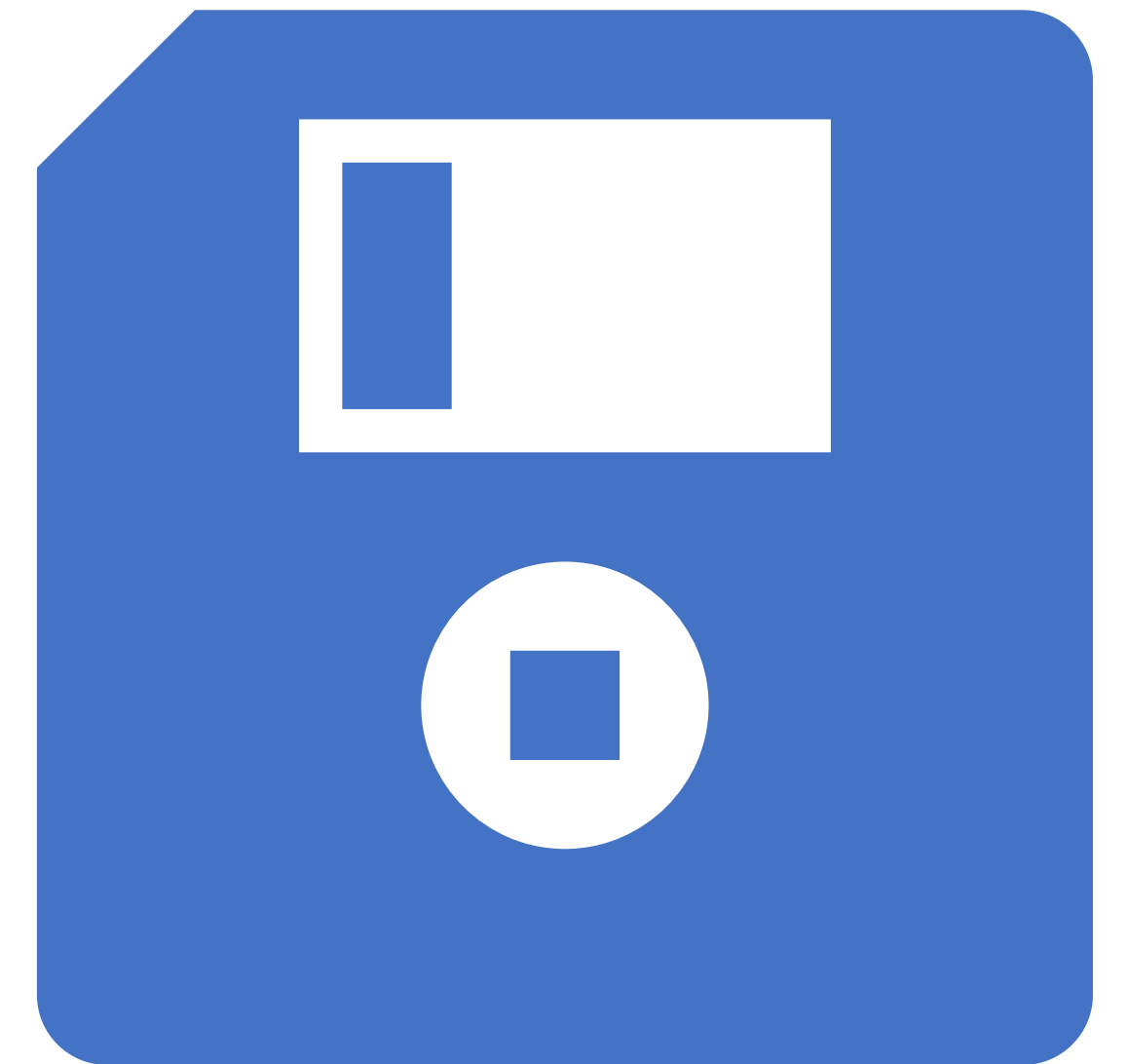


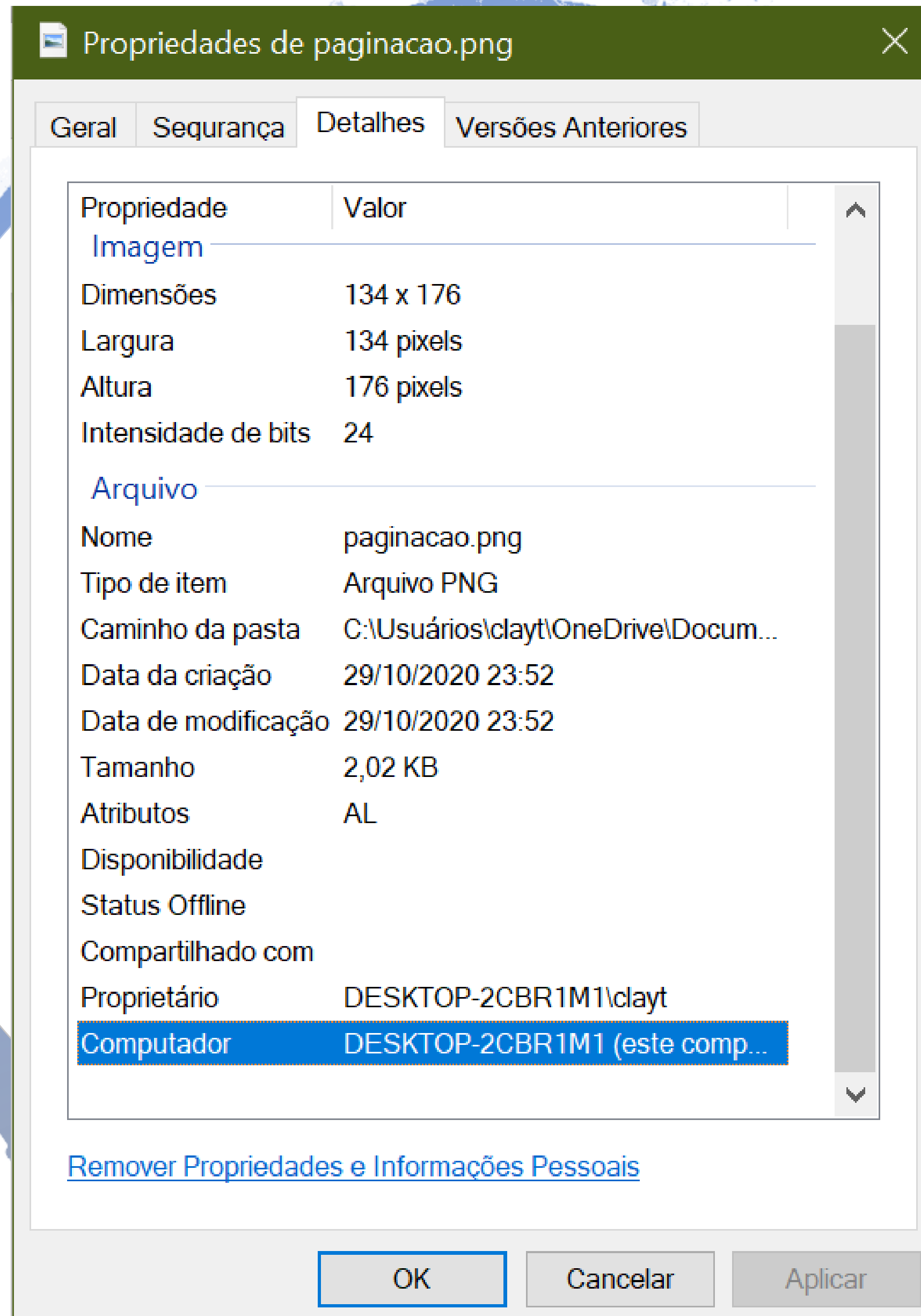
Escalonamento de disco



Gestão de arquivos

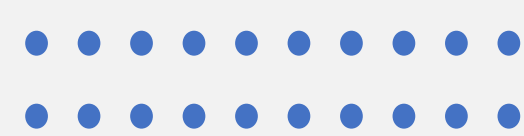
- **Arquivo:** sequência de bytes armazenada em um dispositivo físico não volátil.
- Organizados em **estruturas de dados hierárquicas** denominadas **diretórios**, para facilitar sua localização e acesso pelos usuários.





Atributos ou metadados de arquivos

- Nome
- Tipo
- Tamanho
- Datas
- Proprietário (típico de sistemas multiusuários)
- Permissões de acesso
- Localização



Operações de arquivos

- **Criar** – implicar alocar espaço e definir atributos
- **Abrir** – implica verificar existência, permissões, localização de seu conteúdo e criação de referência em memória
- **Ler/Escrever** – Transferir dados para/de uma área de memória
- **Fechar** – Liberar as estruturas de gerência
- **Remover** – Eliminar o arquivo do dispositivo
- **Alterar atributos** – Modificar os valores dos atributos

Permissões para SISTEMA	Permitir	Negar
Controle total	✓	
Modificar	✓	
Ler & executar	✓	
Leitura	✓	
Gravar	✓	
Permissões especiais		

Para permissões especiais ou configurações avançadas, clique em Avançadas.

Avançadas

Formato de arquivos

- Sequência de bytes
- Arquivos de **registros**
- Arquivos de texto
- Arquivos de código – programa executável ou biblioteca, contém código, tabela de símbolos (variáveis e funções), lista de dependências
- Identificação de conteúdo – extensão do nome

Interface de acesso

- Através dessa interface, um processo pode operar sobre o arquivo
- Composta por uma representação lógica do arquivo, denominada **descriptor de arquivo** (*file descriptor ou file handle*), e um conjunto de funções para manipular o arquivo.
- Níveis de interface de acesso:
 - baixo nível, oferecida pelo sistema operacional aos processos através de chamadas de sistema, e
 - alto nível, composta de funções na linguagem de programação para implementar cada aplicação. Portabilidade de programas entre sistemas operacionais distintos.
- A interface de alto nível é implementada sobre a interface de baixo nível, geralmente na forma de uma biblioteca de funções e/ou um suporte de execução (*runtime*).

Tabela 23.1: Chamadas de sistema para arquivos

Operação	Linux	Windows
Abrir arquivo	OPEN	NtOpenFile
Ler dados	READ	NtReadRequestData
Escrever dados	WRITE	NtWriteRequestData
Fechar arquivo	CLOSE	NtClose
Remover arquivo	UNLINK	NtDeleteFile
Criar diretório	MKDIR	NtCreateDirectoryObject

Tabela 23.2: Funções de biblioteca para arquivos (fd: *file descriptor*, obj: objeto)

Operação	C (padrão C99)	Java (classe File)
Abrir arquivo	fd = fopen(...)	obj = File(...)
Ler dados	fread(fd, ...)	obj.read()
Escrever dados	fwrite(fd, ...)	obj.write()
Fechar arquivo	fclose(fd)	obj.close()
Remover arquivo	remove(...)	obj.delete()
Criar diretório	mkdir(...)	obj.mkdir()



IBMEC.BR

 /IBMEC

 IBMEC

 @IBMEC_OFICIAL

 @IBMEC

 **ibmec**