

FSO

Funções do sistema operativo

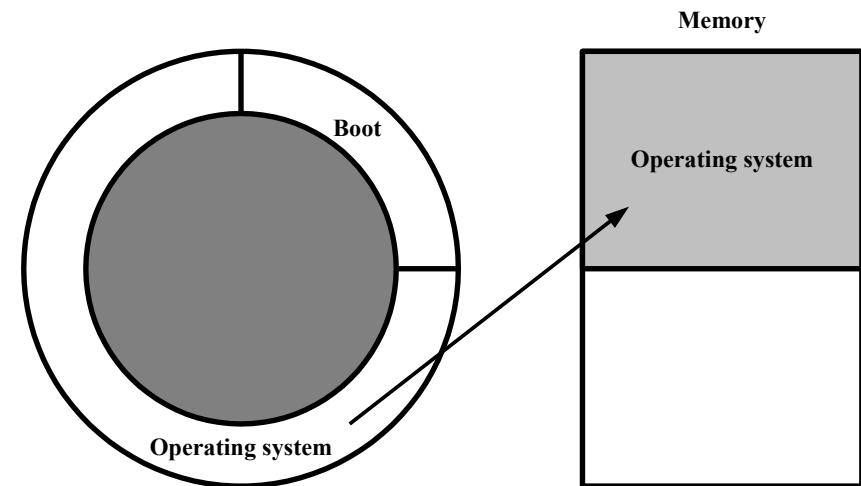
- Virtualização do CPU e da RAM

Bibliografia: livro Operating System: Three Easy Pieces (OSTEP) Secções 2.1, 2.2

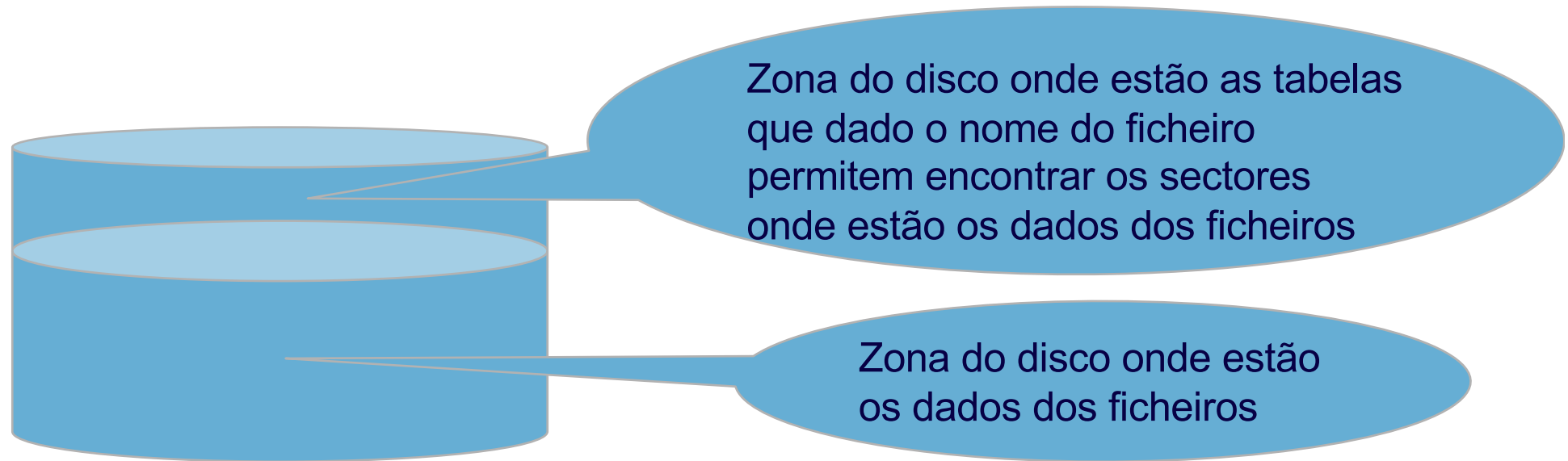
SO é um programa carregado na RAM quando ocorre o “boot” do sistema

1) O “bootstrap loader em ROM” lê o código do SO para RAM

2) O PC do CPU é carregado com o endereço da 1ª instrução do SO



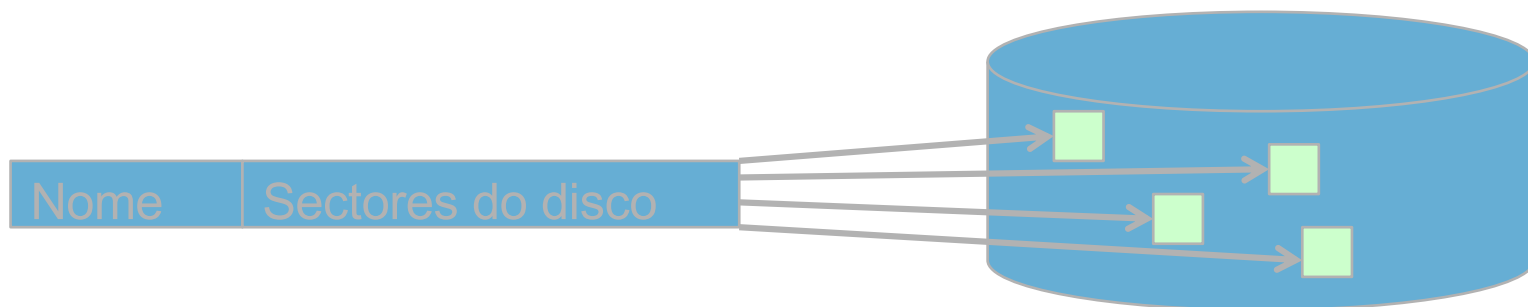
SO COMO GESTOR DE RECURSOS



Se um programa utilizador pudesse fazer acesso livre ao disco ele poderia destruir as zonas do disco em que o SO guarda as tabelas que a partir do nome permitem encontrar os sectores onde estão os dados que pertencem ao ficheiro

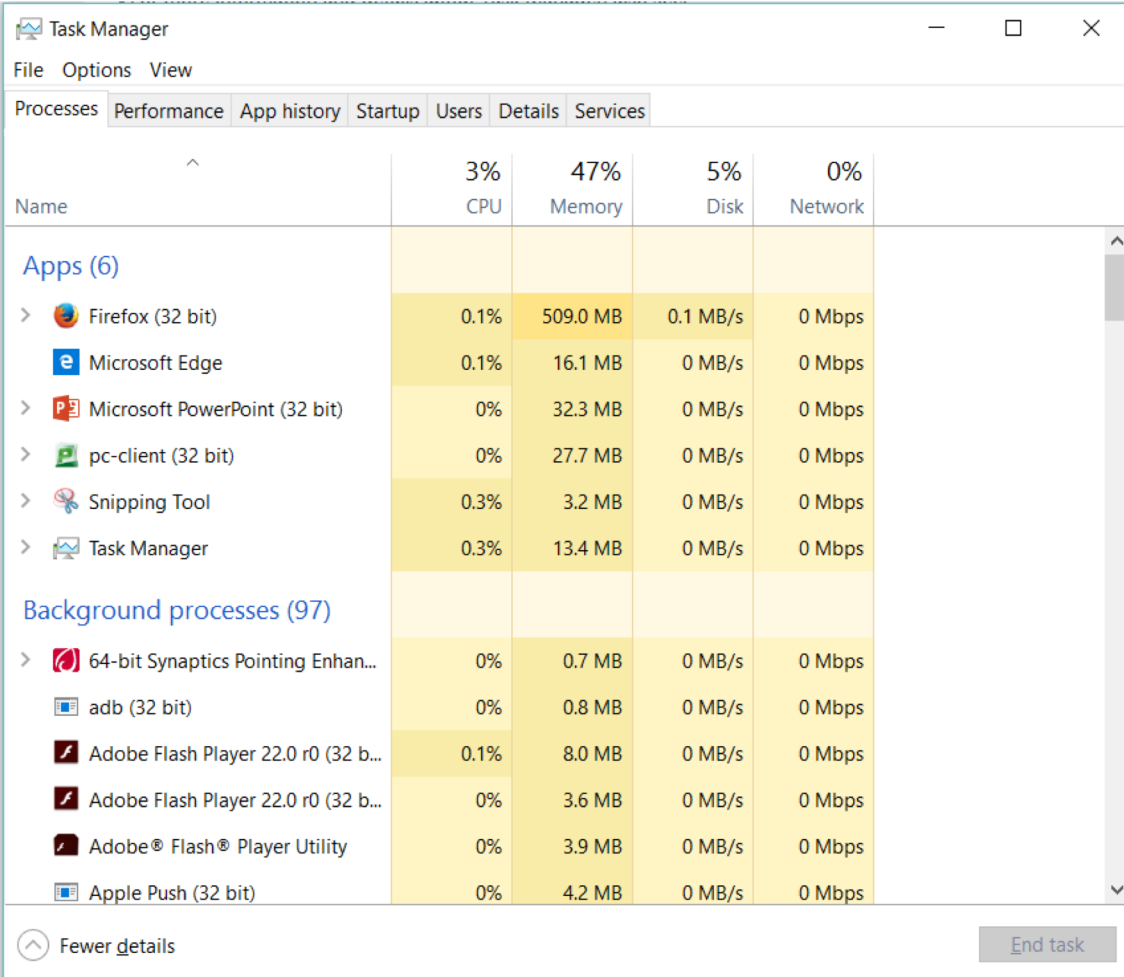
Papel do Sistema de Operação

- Pode ser visto como *um gestor de recursos* – o S.O. é responsável por garantir uma exploração eficiente dos recursos; impede os programas de aplicação de fazer acesso directo aos recursos
- Exemplo, o sistema gere o disco onde guarda os ficheiros
 - Tabela que, para cada ficheiro no disco lhe associa os sectores do disco onde está guardado



FUNÇÕES DO SISTEMA OPERATIVO

- Suportar a execução dos programas
 - Eventualmente muitos ao mesmo tempo



The screenshot shows the Windows Task Manager window with the 'Performance' tab selected. It displays a table of running processes, categorized into 'Apps (6)' and 'Background processes (97)'. The table columns are Name, CPU, Memory, Disk, and Network. The 'Apps' section lists Firefox (32 bit), Microsoft Edge, Microsoft PowerPoint (32 bit), pc-client (32 bit), Snipping Tool, and Task Manager. The 'Background processes' section lists 64-bit Synaptics Pointing Enhancer, adb (32 bit), Adobe Flash Player 22.0 r0 (32 bit), Adobe Flash Player 22.0 r0 (32 bit), Adobe® Flash® Player Utility, and Apple Push (32 bit). The CPU usage is 3%, Memory is 47%, Disk is 5%, and Network is 0%.

Name	CPU	Memory	Disk	Network
Apps (6)				
Firefox (32 bit)	0.1%	509.0 MB	0.1 MB/s	0 Mbps
Microsoft Edge	0.1%	16.1 MB	0 MB/s	0 Mbps
Microsoft PowerPoint (32 bit)	0%	32.3 MB	0 MB/s	0 Mbps
pc-client (32 bit)	0%	27.7 MB	0 MB/s	0 Mbps
Snipping Tool	0.3%	3.2 MB	0 MB/s	0 Mbps
Task Manager	0.3%	13.4 MB	0 MB/s	0 Mbps
Background processes (97)				
64-bit Synaptics Pointing Enhancer	0%	0.7 MB	0 MB/s	0 Mbps
adb (32 bit)	0%	0.8 MB	0 MB/s	0 Mbps
Adobe Flash Player 22.0 r0 (32 bit)	0.1%	8.0 MB	0 MB/s	0 Mbps
Adobe Flash Player 22.0 r0 (32 bit)	0%	3.6 MB	0 MB/s	0 Mbps
Adobe® Flash® Player Utility	0%	3.9 MB	0 MB/s	0 Mbps
Apple Push (32 bit)	0%	4.2 MB	0 MB/s	0 Mbps

Funções do SO

- O sistema de operação (SO) é o componente do software de sistema que serve de suporte a todo software de aplicação e de sistema
- Os SOs controlam a execução de programas, a gestão de recursos, protecção e segurança
- O SO cria uma **máquina virtual com instruções de alto nível** que os programas podem usar fazendo chamadas ao sistema
- O **SO impede o acesso directo aos periféricos e ficheiros** pelos programas

EXECUTAR PROGRAMAS, COMO ?

É preciso oferecer a cada programa uma máquina com

- *CPU*: para executar instruções
- *RAM*: para guardar código dados e pilha
- *Canais de entrada e saída*: Formas de comunicar com o exterior

EXECUTAR PROGRAMAS, COMO ?

- **Virtualização:** O sistema operativo cria as **máquinas virtuais** para cada programa executar a partir dos recursos físicos
 - **CPUs**
 - **RAM**
 - **Dispositivos de entrada saída** (discutido mais tarde)
- **Partilha de recursos:** Os vários programas em execução partilham os recursos, sob a supervisão do S.O.

Duas Abstracções Suportadas pelo SO

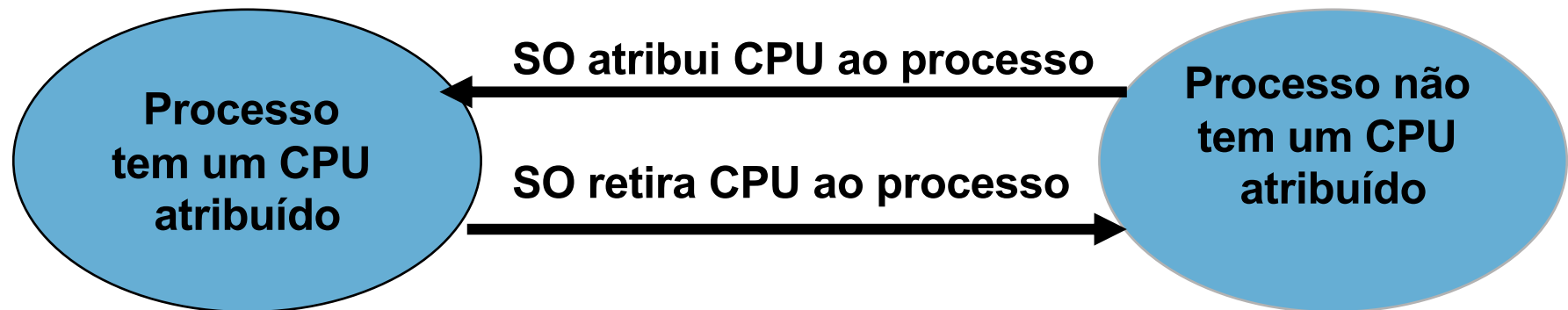
- **Ficheiro**
 - Disponibilizado ao programador como forma de esconder os detalhes de manipulação da memória persistente (discos)
- **Processo**
 - Disponibilizado ao programador como forma de esconder os detalhes da gestão de recursos necessários à execução de programas

Processo

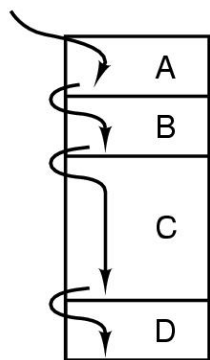
- **SO suporta para cada processo**
 - Um **CPU virtual** suportado na partilha do tempo do CPU (ou CPUs) reais
 - Um **conjunto de memória** (imagem do processo) obtida por partilha espacial da memória
 - Um conjunto de canais de entrada-saída que permitem enviar e receber bytes
 - De/Para Periféricos
 - De/Para Ficheiros
 - De/Para Outros processos

VIRTUALIZAÇÃO DO(s) CPU(s)

A repartição do CPU é feita temporalmente, sendo atribuídos a cada processo um CPU físico durante alguns milissegundos consecutivos



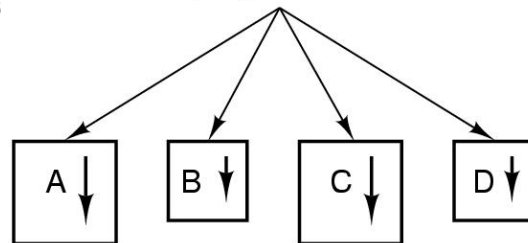
One program counter



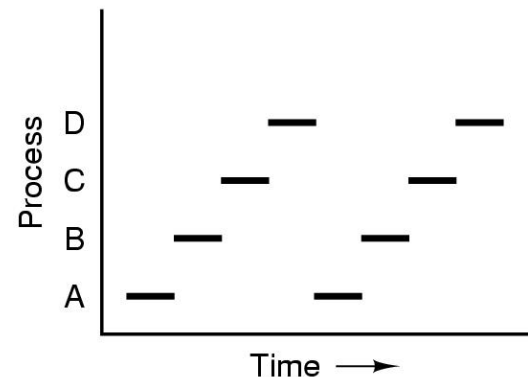
(a)

Process switch

Four program counters



(b)

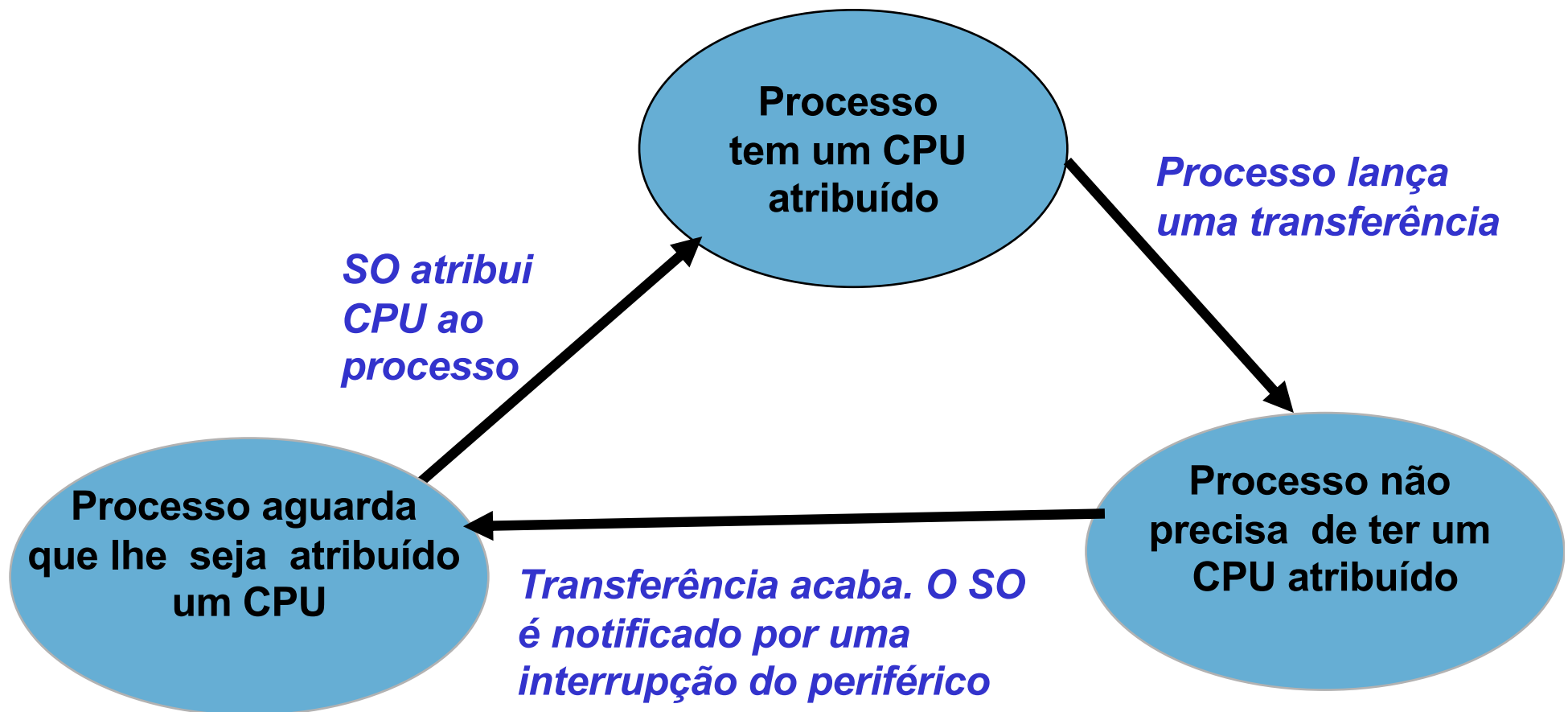


(c)

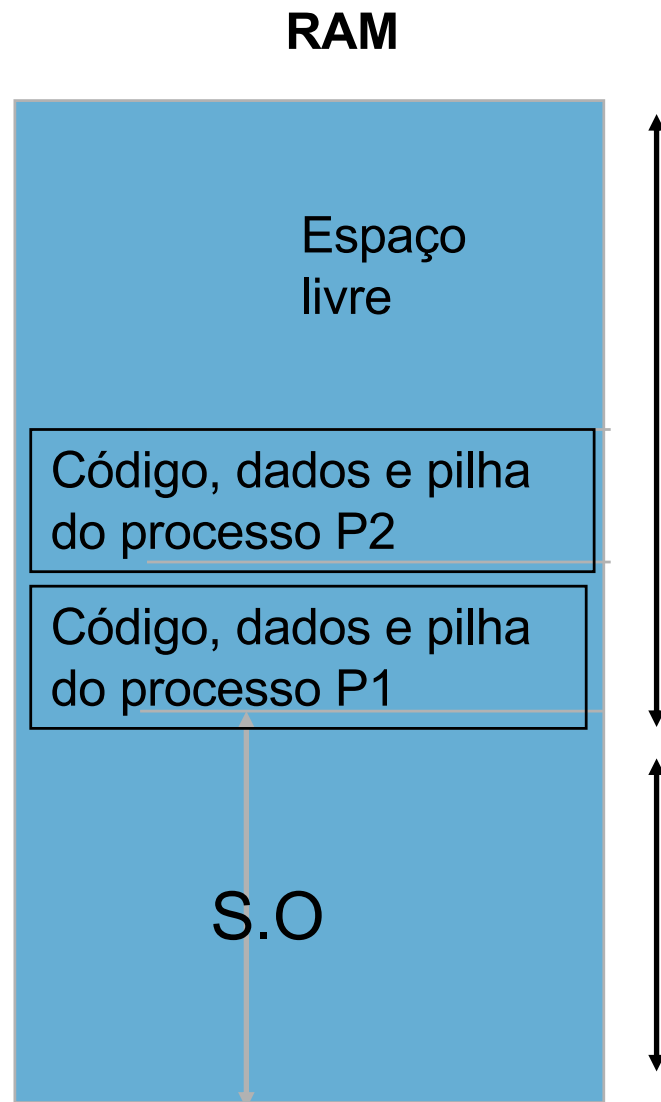
Tanenbaum,
Modern Operating
Systems

VIRTUALIZAÇÃO DO(s) CPU(s)

A repartição do CPU é feita tirando partido do facto de os periféricos serem muito lentos e os processos passarem muito à espera que transferências de informação acabem



VIRTUALIZAÇÃO DA RAM



A repartição da RAM é feita espacialmente, sendo atribuídos aos processos faixas de endereços contíguos

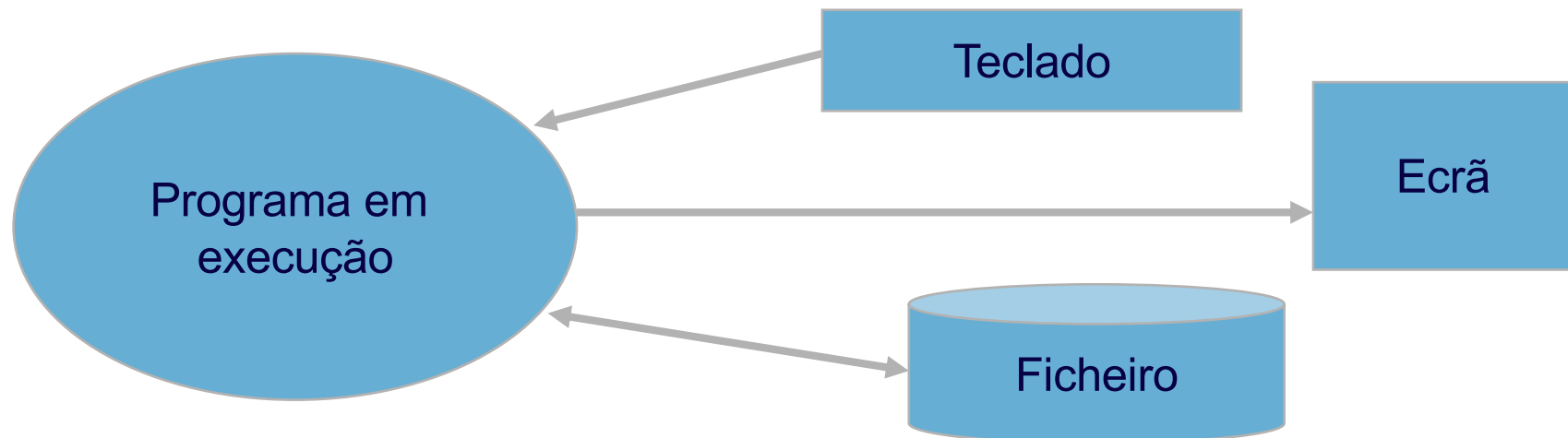
RAM disponível para os processos. Quando um processo é lançado ocupa memória; quando termina a memória é libertada

RAM ocupada pelo SO

Canais de entrada/saída

Um programa em execução vê periféricos e ficheiros através de **canais de entrada saída**

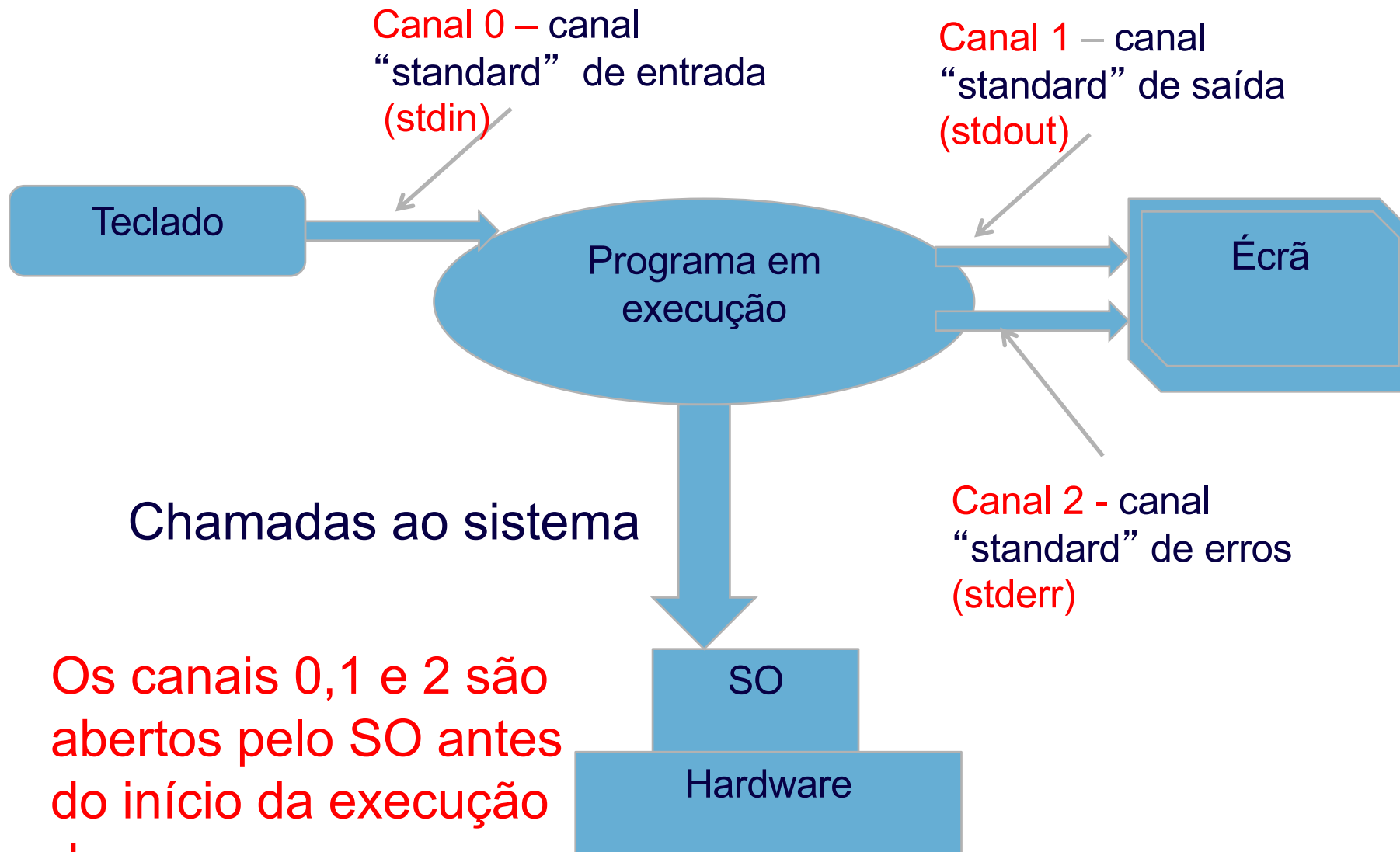
Esses canais permitem ler e escrever bytes de forma fiável e mantendo a ordem



Os canais são identificados por números: começam em 0 e vão crescendo à medida das necessidades

- Os **canais 0 (stdin), 1 (stdout) e 2 (stderr)** são abertos pelo SO quando o programa começa a executar
- o programa pode abrir mais canais usando a chamada **open**

Programa em execução



FICHEIRO

- A informação é guardada de forma permanente.
- Um ficheiro é um espaço de endereçamento logicamente contíguo, contendo um conjunto de dados “inter-relacionados” e é acessível através de um **identificador único**
- Um ficheiro pode conter dados (texto, imagem, som, etc.) ou programas.
- O SO encarrega-se de gerir os ficheiros e os discos em que estes residem. A organização do disco é escondida aos programadores e utilizadores.