

# 1. Arquitetura do sistema (visão geral)

## 1.1. Conceitos básicos

Um Sistema Operativo, SO (*Operating System* – OS), é o **software principal de um computador**, pois sem este um computador não funciona. O sistema operativo cria uma interligação entre o *hardware* do computador e o utilizador. Permite também que outro *software* funcione interagindo com o *hardware* instalado, para que possa executar funções, programar e abrir aplicações.

CITIGASIS16 © Porto Editora



Fig. 5.1 Comparação do sistema operativo a um guarda de trânsito

Analisemos a seguinte situação.

O trânsito num cruzamento do centro de uma grande cidade pode ser comparado ao que se passa no interior de um computador. Os eletrões vibram a uma grande velocidade, deslocando-se de um lado para outro, equivalente eletrónico dos carros. Os programas e periféricos impacientes tocam “buzinas” eletrónicas, tentando atrair a atenção do guarda para os poder atender.

O sistema operativo do computador é responsável por manter o trânsito fluido. Podes imaginar o sistema operativo como uma versão computadorizada de um guarda de trânsito, localizado no cruzamento entre o *hardware* do computador, os programas de aplicações e o utilizador (ver figura 5.1).

Resumindo, podemos dizer que a missão de um sistema operativo é a de criar uma máquina virtual sobre a máquina física que ofereça os recursos lógicos básicos necessários ao desenvolvimento das aplicações.



Fig. 5.2 Sistema operativo

Vejamos alguns conceitos associados aos sistemas operativos.

### 1.1.1. Núcleo de SO

O sistema operativo trabalha estritamente com o *hardware* do computador e com os programas aplicativos; todos estes componentes têm de ser projetados para trabalharem em conjunto. A parte do sistema operativo que faz a interligação com o *hardware* chama-se **núcleo (kernel)**.

O *software* aplicativo é desenvolvido para funcionar num sistema operativo específico. Um programa que funcione num iMac requer o sistema operativo Mac OS e não

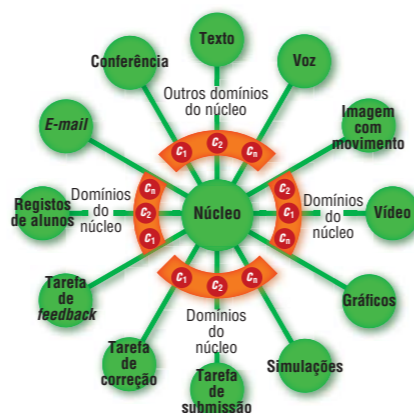


Fig. 5.3 Núcleo (kernel)

poderá ser executado noutra sistema operativo, como, por exemplo, no Windows 7. Existem programas para Windows que poderão ser executados nos computadores da Apple, ou Linux, mas para tal é necessário um programa de emulação do Windows. Atualmente, são desenvolvidos *softwares* para correrem em diferentes sistemas operativos. Por exemplo, no Microsoft Office existe uma versão que funciona nos sistemas operativos da Microsoft (Windows) e outra versão para correr sobre o sistema operativo Mac OS. Outro exemplo, ainda mais abrangente, é o caso do OpenOffice que é um produto concorrente do Microsoft Office e que é desenvolvido pela Sun. Este, apesar de ser gratuito, contém versões para funcionarem sobre o Windows, Unix, Linux e OS/2, entre outros.

### 1.1.2. Processos

Um conceito-chave da teoria dos sistemas operativos é o conceito de processo. Um processo é basicamente um programa em execução. Podes obter a lista dos processos em execução no SO Windows XP pressionando as teclas CTRL + ALT + DEL.

Cada processo possui um espaço de endereçamento para leitura e/ou escrita. O **espaço de endereçamento** do processo contém o programa executável, os dados do programa e a pilha (*stack*) associada. A pilha é uma memória especial utilizada na execução de rotinas do núcleo. Podem estar associados ao processo alguns registos de *hardware* com todas as informações necessárias à execução ou paragem de um programa.

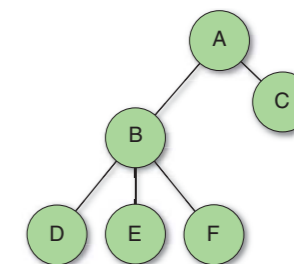


Fig. 5.4 Árvore de processos

### 1.1.3. Chamadas ao sistema

Uma preocupação que surge na maior parte dos projetos de sistemas operativos é a implementação de mecanismos de proteção do núcleo e do acesso aos seus serviços. Caso uma aplicação, com acesso ao núcleo, realize uma operação que o danifique, todo o sistema poderá ficar comprometido e inoperante.

O utilizador (ou processo), quando deseja solicitar algum serviço do sistema, realiza uma chamada a uma das suas rotinas (ou serviços) através de chamadas ao sistema (**system calls**). As chamadas ao sistema são a porta de entrada para se ter acesso ao núcleo do sistema operativo.

Para cada serviço existe uma chamada ao sistema associado; cada sistema operativo tem o seu próprio conjunto (biblioteca) de chamadas, com nomes, parâmetros e formas específicas de ativação.

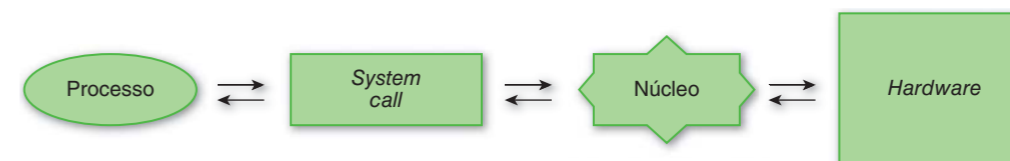


Fig. 5.5 Chamada ao sistema (system call)