



Sistemas de Operação

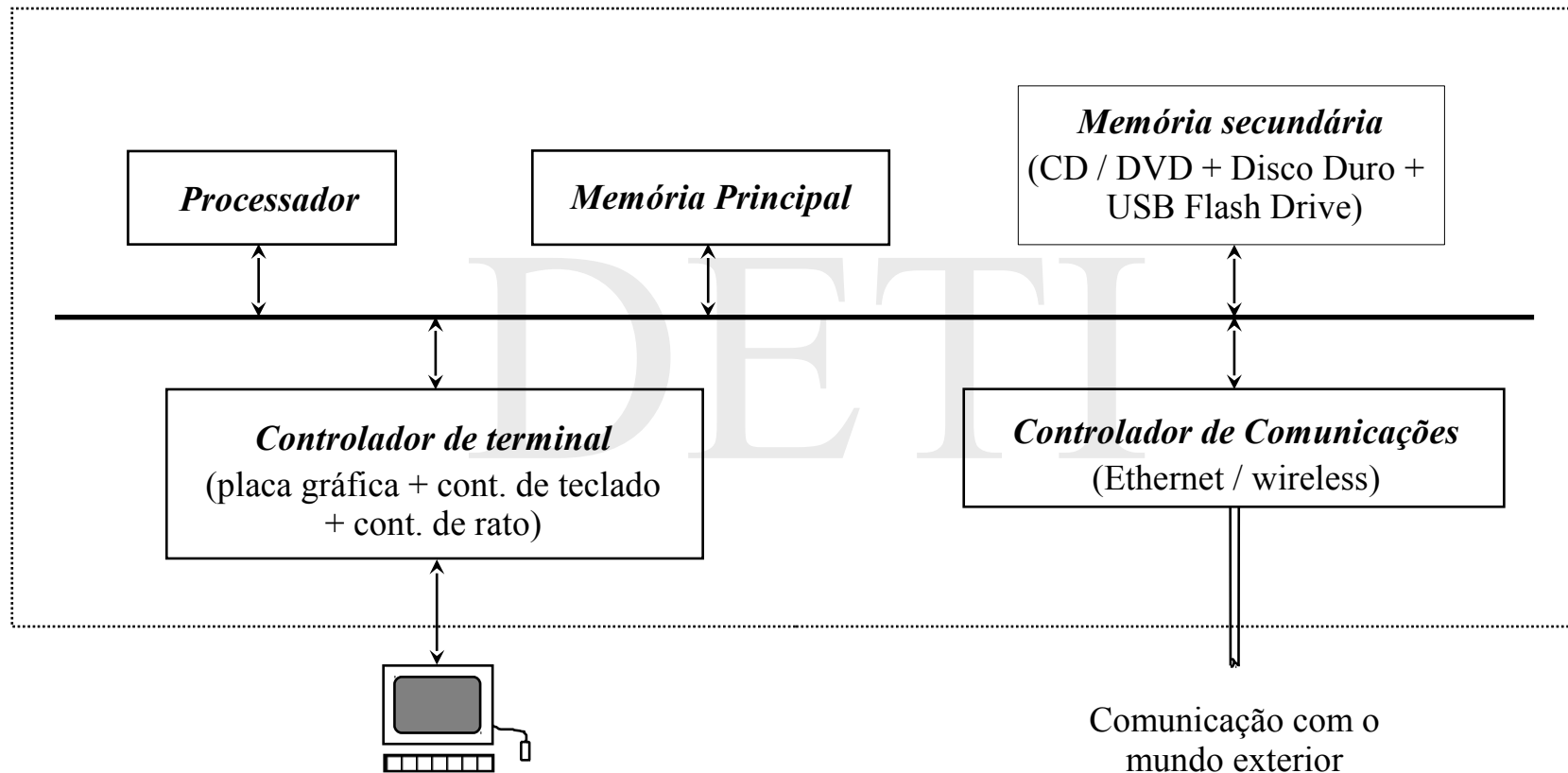
Conceitos Introdutórios

António Rui Borges / Artur Pereira

Sumário

- ♦ *Princípios gerais*
- ♦ *Evolução dos sistemas de operação*
- ♦ *Tipos de ambientes de processamento*
- ♦ *Características comuns aos sistemas de operação atuais*
- ♦ *Arquitetura interna*
- ♦ *Leituras sugeridas*

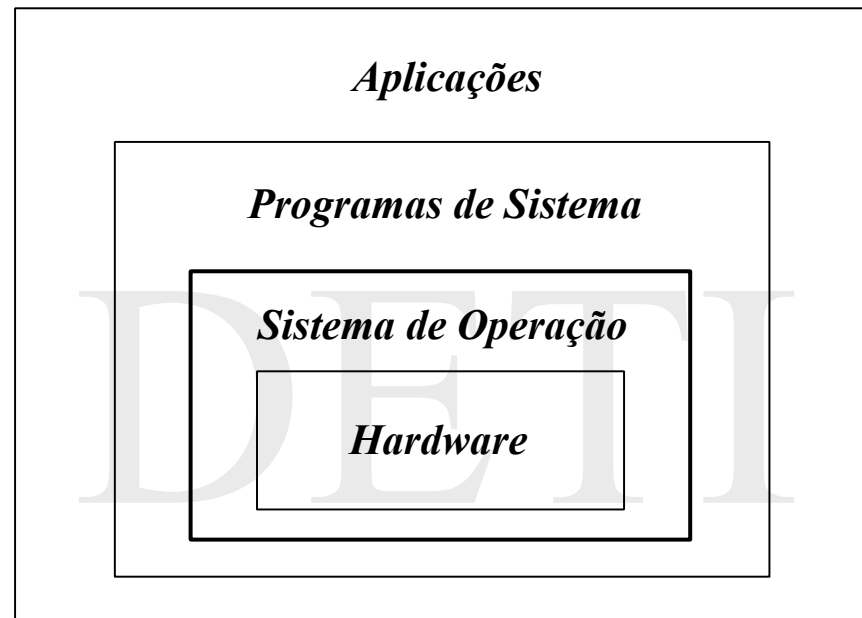
Sistema computacional típico



Sistema de Operação: visão simplificada

- ♦ programa base executado pelo sistema computacional;
- ♦ programa que dá 'vida' ao computador (*hardware*), possibilitando a criação de um ambiente de interação com o(s) utilizador(es) para realização de trabalho útil sobre a máquina (edição, compilação, 'linkagem' e execução de progra-mas, etc.);
- ♦ pode ser de dois tipos:
 - ♦ **gráfico** - baseia-se na construção de janelas sobre o ecrã do monitor vídeo que atuam como centros de comunicação entre o utilizador e aplicações específicas; elementos gráficos variados, designados de *icons*, são aí desenhados e podem ser manipulados usando o rato;
 - ♦ **linguagem de comandos (shell)** - baseia-se em linhas de texto introduzidas pelo teclado que formam os comandos a serem executados pelo ambiente; tipicamente, existe uma linguagem de programação que possibilita uma abordagem mais estruturada à construção de comandos complexos.

Enquadramento geral



- ♦ Duas perspetivas:
 - ♦ Top-down ou do Programador
 - ♦ Bottom-up ou do Construtor

Perspetiva 'top-down' (ou do programador) - 1

- ♦ o **sistema de operação** transmite ao programador uma **abstração do sistema computacional** que o liberta do conhecimento preciso dos detalhes do *hardware* subjacente:
 - ♦ fornece um modelo funcional do sistema computacional, designado pelo nome de *máquina virtual*, que é mais simples de compreender e de programar;
 - ♦ o interface com o *hardware*, assim criado, origina um ambiente uniforme de programação que é operacionalizado através de **chamadas ao sistema** e que possibilita, por isso, a portabilidade de aplicações entre sistemas computacionais estruturalmente distintos.

Perspectiva 'top-down' (ou do programador) - 2

- ♦ funcionalidades criadas pelo sistema de operação:
 - ♦ estabelecimento do ambiente base de interacção com o utilizador;
 - ♦ disponibilização de facilidades para o desenvolvimento, teste e validação de programas;
 - ♦ fornecimento de mecanismos para a execução controlada de programas, sua comunicação e sincronização mútuas;
 - ♦ dissociação do espaço de endereçamento do programa das limitações impostas pelo tamanho da memória principal;
 - ♦ organização da memória secundária em sistemas de ficheiros;
 - ♦ definição de um modelo geral de acesso aos dispositivos de entrada / saída, independente das suas especificidades próprias;
 - ♦ deteção de situações de erro e estabelecimento da resposta adequada.

Perspetiva 'bottom-up' (ou do construtor)

- ♦ Pode definir-se **sistema computacional** como sendo um sistema formado por um conjunto de recursos (processador(es), memória principal, memória secundária e diferentes tipos de controladores de dispositivos de entrada/saída) para processamento e armazenamento de informação.
- ♦ Neste sentido, o sistema de operação é visto como
 - ♦ o programa que gere o sistema computacional, fazendo a atribuição controlada e ordeira dos seus diferentes recursos aos programas que por eles competem;
 - ♦ e que visa a rentabilização máxima do sistema computacional, garantindo uma utilização tão eficiente quanto possível dos recursos existentes.

Evolução dos sistemas de operação - 1

Caracterização tecnológica

Observações

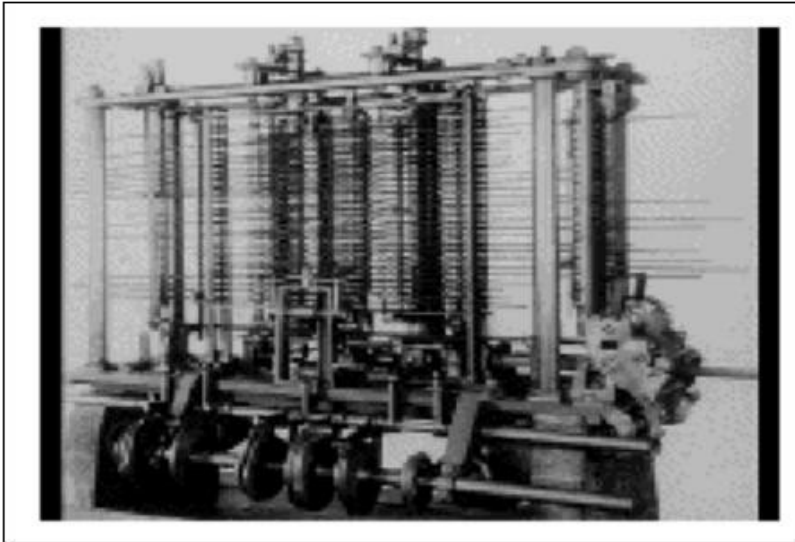
Pré-História

(meados do século XIX)

Charles Babbage
Lady Ada Lovelace

máquina analítica
(dispositivo mecânico)

[a tecnologia da altura não permitiu
a sua efectiva construção]
• *não tinha sistema de operação*



Máquina analítica
Charles Babbage

Evolução dos sistemas de operação - 2

Caracterização tecnológica

Observações

Geração 1

(1945 -1955)

John von Neumann

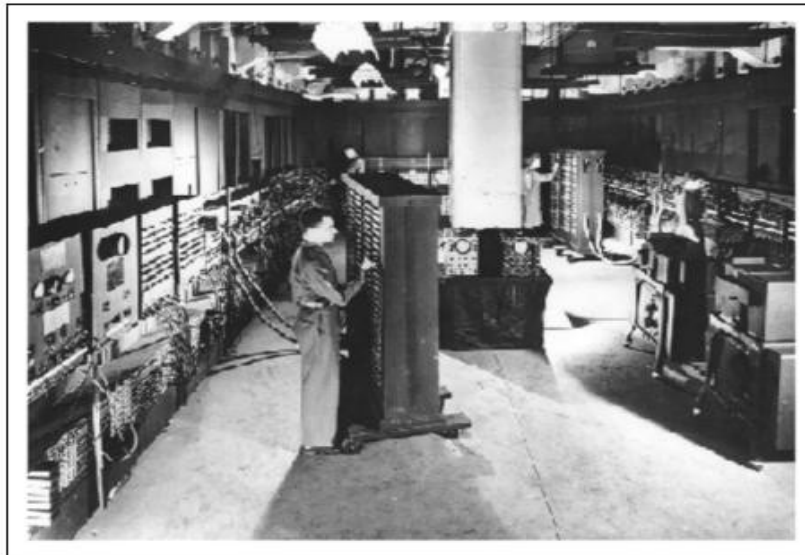
Howard Aiken

Eckert / Mauchley

Konrad Zuse

dispositivos a
válvulas eletrónicas

- *não tinha sistema de operação*
- a programação era feita em linguagem máquina
- o programador controlava completamente a máquina
- o processamento era feito em série



ENIAC
Eckert e Mauchley
Univ. da Pensilvânia

Evolução dos sistemas de operação - 3

Caracterização tecnológica

Observações

Geração 2 (1955 -1961)

dispositivos a
a transistores

- sistema de operação de tipo *batch* simples
- introdução do conceito de monitor e de uma linguagem de comandos rudimentar (*job control language*)

Geração 3 (1961 -1978)

dispositivos a circuitos
integrados (SSI-MSI-LSI)

noção de família de computadores
(IBM 360)
aparecimento do minicomputador
invenção do microprocessador

- sistema de operação de tipo *batch* multiprogramado
- introdução dos sistemas interativos (*time-sharing*)
- sistemas de operação de uso geral (*proprietários*) de grande complexidade
- sistemas de tempo real
- surgimento do *Unix*

Evolução dos sistemas de operação - 4

Caracterização tecnológica

Observações

Geração 4 (1978 - 1993)

dispositivos a circuitos
integrados (VLSI)
popularização da estação de
trabalho e do computador pessoal
multiprocessadores
divulgação da tecnologia de
comunicações (redes de computadores)

- implantação de sistemas de operação *standard* (*MS_DOS, Windows, Unix*)
- sistemas de operação de rede com integração progressiva de facilidades de comunicação

Geração 5 (1993 - ...)

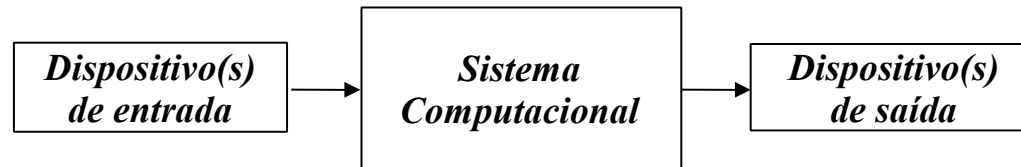
ligação progressivamente mais
íntima entre os sistemas computacionais
e a tecnologia de comunicações
(banda larga, wireless)
hardware reconfigurável
dinamicamente
SMP on chip
exploração de novos paradigmas tecnológicos
(computadores biológicos, computadores
ópticos, ...)

- sistemas distribuídos
- internet
- system on chip
- computação grid
- computação móvel
- computação ubíqua
- novos interfaces de interacção
- *cloud computing*

Tipos de sistemas de operação

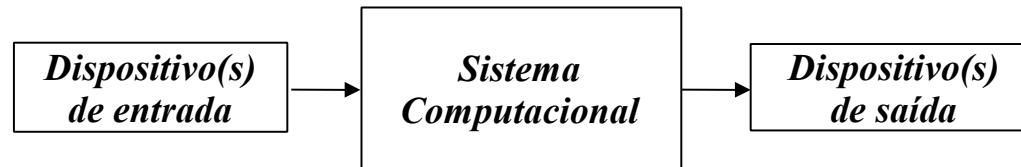
- ♦ Os sistemas de operação podem ser classificados de acordo com o modo como o processamento é feito:
 - ♦ Processamento em série
 - ♦ Processamento em *batch* simples
 - ♦ Processamento em *batch* multiprogramado
 - ♦ Sistemas interativos (*time-sharing*)
 - ♦ Sistemas de tempo real
 - ♦ Sistemas de rede

Processamento em série



- ♦ **Método:** o programador/operador tem um **controle completo** sobre o sistema computacional, executando sucessivamente programas de sistema que lhe permitem o desenvolvimento e teste do seu programa (edição, compilação, 'linkagem' e carregamento em memória para execução);
- ♦ **Dispositivos de entrada típicos:** leitor de cartões (ou de fita perfurada), unidades de banda magnética.
- ♦ **Dispositivos de saída típicos:** perfurador de cartões (ou de fita), unidades de banda magnética, impressora.
- ♦ o sistema **operativo** é elementar; é formado principalmente por um conjunto de rotinas que **estabelecem** a comunicação com os dispositivos de entrada/saída (*IOCS – input/output control system*).

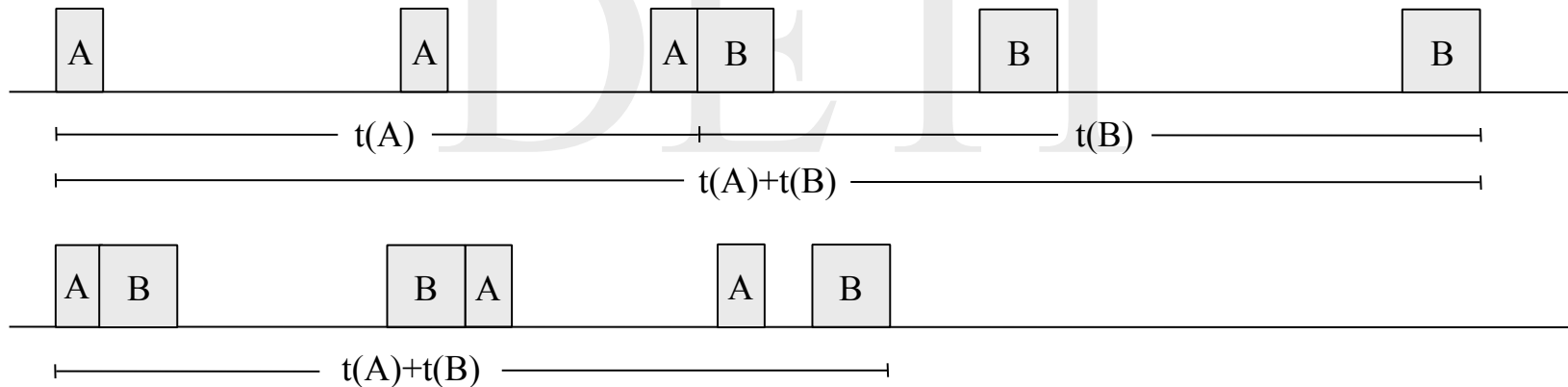
Sistema de tipo batch simples



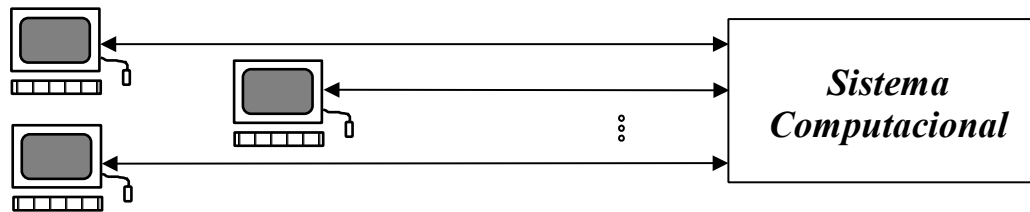
- ♦ **Objectivo:** manter o processador ocupado por minimização da intervenção humana
 - ♦ as operações necessárias ao desenvolvimento e teste dos programas são automatizadas.
- ♦ **Método:** introdução do conceito de *job* (agrupamento de todas as operações relativas ao tratamento de um programa); fornecimento em bloco de um conjunto (*batch*) de *jobs* a serem processados sequencialmente; aparecimento do sistema de operação como um programa monitor que controla o sistema computacional e que interpreta comandos escritos numa linguagem própria (*job control language*):
\$JOB #####, \$FORTRAN #####, \$RUN, \$DATA #####, \$END.
- ♦ **Dispositivos de entrada típicos:** unidades de banda magnética.
- ♦ **Dispositivos de saída típicos:** unidades de banda magnética, impressora.

Sistema de tipo batch multiprogramado

- **Objectivo:** otimizar a ocupação do processador.
- **Método:** aproveitamento dos tempos mortos durante a execução de um programa, enquanto o processador aguarda pela realização das operações de entrada/saída, para a execução de outro programa.

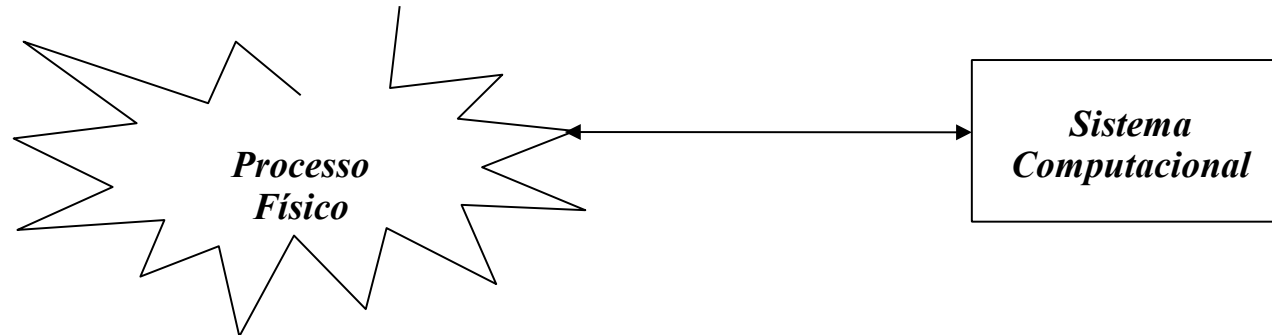


Sistema de tipo interativo ('time-sharing')



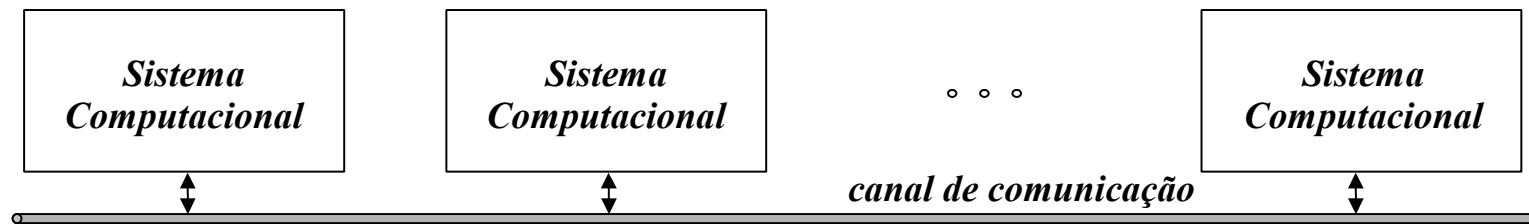
- ♦ **Objectivo:** criação de um ambiente de trabalho confortável para o utilizador através da minimização do tempo de resposta do sistema a solicitações externas.
- ♦ **Método:** manter os diferentes utilizadores, cada um ligado ao seu próprio dispositivo de entrada/saída, em contacto direto e simultâneo com o sistema computacional;
 - ♦ usando a multiprogramação, o processador é sucessivamente atribuído à execução dos diversos programas presentes durante intervalos de tempo muito curtos, *time-slots*;
 - ♦ face à lentidão do tempo de resposta humano, quando comparado com um computador, cria-se a ilusão nos utilizadores que o sistema lhes está inteiramente dedicado.
- ♦ **Dispositivos de entrada típicos:** terminais.
- ♦ **Dispositivos de saída típicos:** terminais, impressora.

Sistema de operação de tempo real



- ♦ **Objectivo:** utilizar o computador na monitorização e controlo *on-line* de processos físicos.
- ♦ **Método:** variante de um sistema interativo, onde se procura impor limites máximos aos tempos de resposta do sistema computacional a diferentes classes de solicitações externas.

Sistema de operação de rede



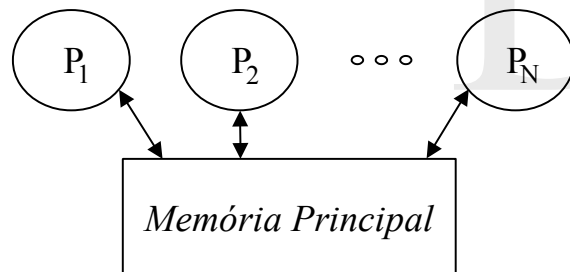
- ♦ **Objetivo:** tirar partido das facilidades de interligação de sistemas computacionais (ao nível de *hardware*) para estabelecer um conjunto de serviços comuns a toda uma comunidade.
- ♦ **Serviços:** transferência de ficheiros entre sistemas computacionais (*ftp*); partilha de sistemas de ficheiros remotos, criando a ilusão de que são locais (*NFS*); partilha de dispositivos remotos (impressoras, etc.); acesso a sistemas computacionais remotos (*telnet*, *remote login*, *ssh*); correio eletrónico (*e-mail*); acesso à Internet (*browsers*).
- ♦ **Dispositivos de comunicação típicos:** Ethernet, wireless.

Sistema de operação distribuído

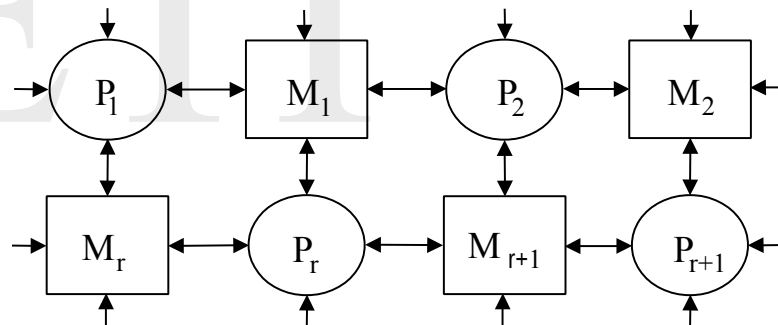
- ♦ **Objetivo:** tirar partido das facilidades de construção de sistemas computacionais com processadores múltiplos, ou de interligação de sistemas computacionais distintos, para estabelecer um ambiente integrado de interação com o(s) utilizador(es) que encara o sistema computacional paralelo como uma entidade única.
- ♦ **Metodologia:** garantir uma transparência tão completa quanto possível no acesso aos processadores, ou aos diferentes recursos dos sistemas computacionais interligados, tendo em vista
 - ♦ o balanceamento estático e/ou dinâmico de carga;
 - ♦ o aumento da velocidade de processamento por incorporação de novos processadores ou sistemas computacionais;
 - ♦ a paralelização de aplicações;
 - ♦ a implementação de mecanismos de tolerância a falhas.

Multiprocessamento vs. Multiprogramação - 1

- ♦ **Paralelismo** - é a capacidade apresentada por um sistema computacional de poder executar em simultâneo dois ou mais programas
 - ♦ exige que o sistema computacional seja formado por mais do que um processador (um por cada programa em execução simultânea).
 - ♦ Quando tal acontece, diz-se que o sistema operativo associado tem características de **multiprocessamento**.



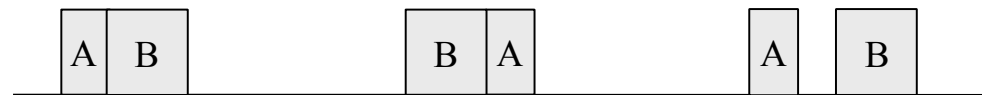
*processamento simétrico
(SMP)*



malha planar

Multiprocessamento vs. Multiprogramação - 2

- ♦ **Concorrência** - é a ilusão criada por um sistema computacional de aparentemente poder executar em simultâneo mais programas do que o número de processadores existentes
 - ♦ exige que a atribuição do(s) processador(es) seja **multiplexada** no tempo entre os diferentes programas presentes.
 - ♦ Quando tal acontece, diz-se que o sistema operativo associado tem características de **multiprogramação**.



- ♦ Num monoprocessador (sistema computacional com um único processador), os programas A e B estão a ser executados em concorrência.

Características comuns aos sistemas de operação atuais

- ♦ apresentam um ambiente gráfico de interação com o utilizador;
- ♦ são sistemas interativos multiutilizador e multitarefa
 - ♦ permitem a interação em simultâneo com múltiplos utilizadores;
 - ♦ permitem que cada utilizador tenha em simultâneo múltiplos programas em execução;
- ♦ implementam uma organização de memória virtual;
- ♦ são sistemas de operação de rede
 - ♦ permitem o acesso de um modo quase indistinto a sistemas de ficheiros e a dispositivos de entrada / saída locais e remotos;
 - ♦ incluem aplicações que permitem, entre outras coisas, o *login* em máquinas remotas, o correio eletrónico e a navegação na Internet;
- ♦ têm uma coleção muito grande de *device drivers* para potenciar a interligação de tipos muito variados de dispositivos de entrada / saída;
- ♦ permitem em muitos casos a ligação dinâmica de dispositivos (*plug and play*).

Arquitetura interna - 1

- ♦ Face às funcionalidades que um sistema de operação deve apresentar, é imediato concluir
 - ♦ que se trata de um programa muito complexo, de tamanho muito grande (pode atingir as dezenas de milhão de instruções)
 - ♦ que exige cuidados enormes na sua conceção e implementação para funcionar corretamente.

Arquitectura interna - 2

- ♦ As abordagens mais comuns são
 - ♦ **abordagem monolítica** – o núcleo (*kernel*) do sistema de operação é concebido como um programa único, em que todas as funcionalidades, e estruturas de dados associadas, são diretamente acessíveis
 - ♦ implementação muito eficiente, mas muito difícil de testar e modificar;
 - ♦ **abordagem modular** – o núcleo (*kernel*) do sistema de operação é concebido numa perspectiva *top-down*, em que as diferentes funcionalidades são identificadas e isoladas umas das outras através da especificação de um interface de comunicação
 - ♦ implementação menos eficiente do que a anterior, mas onde se torna possível o teste separado dos diversos componentes e efetuar, com o mínimo de riscos, alterações parcelares ou introdução de novas funcionalidades.

Arquitectura interna - 3

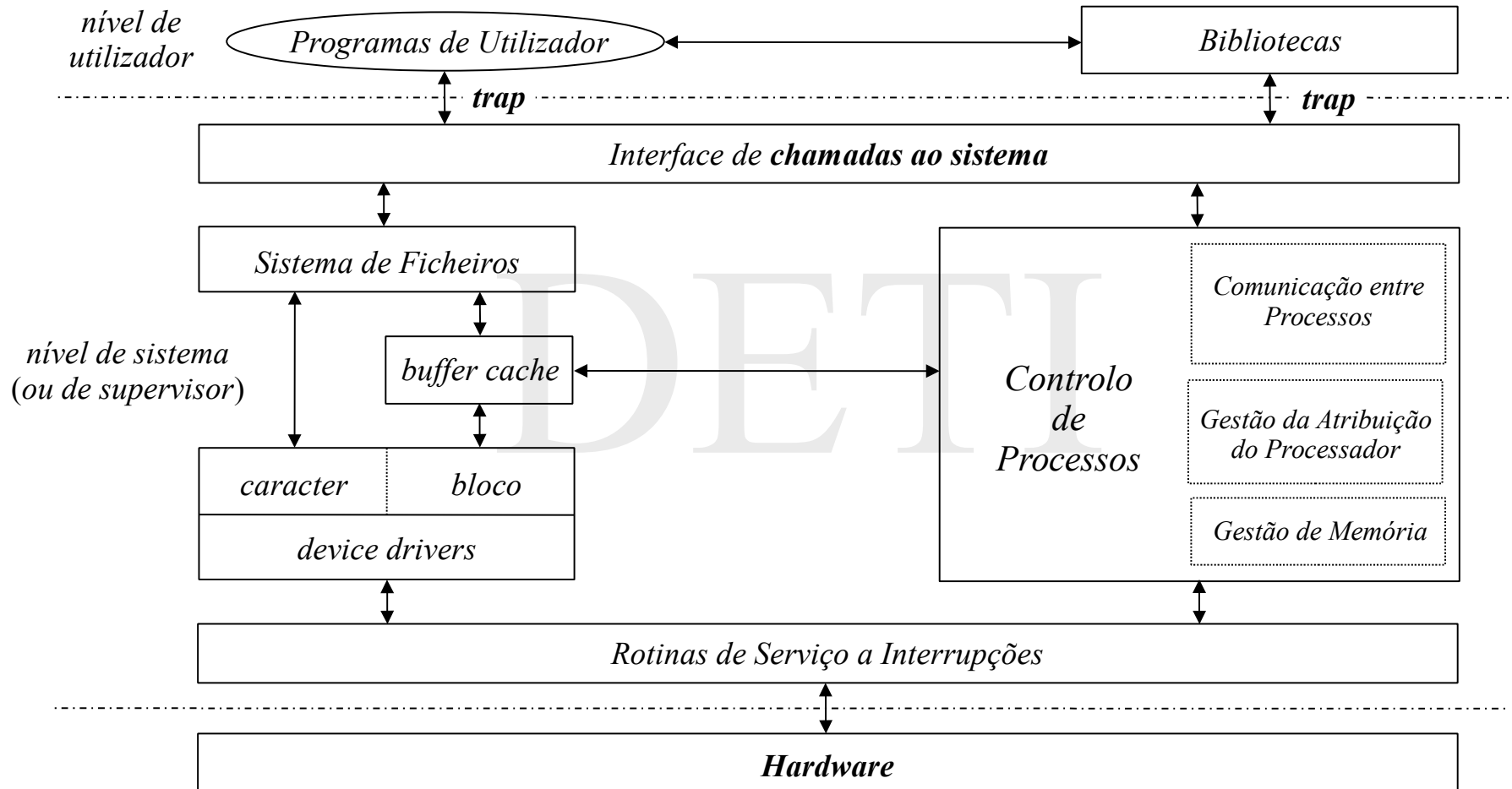
A *abordagem modular* pode, por sua vez, ser enquadrada sob diferentes perspetivas que não são necessariamente mutuamente exclusivas

- ♦ **arquitetura em camadas** – decomposição hierárquica em que cada novo módulo é construído sobre os módulos anteriores usando as operações fornecidas
- ♦ embora conceptualmente seguro, exige uma cuidadosa definição das funcionalidades de cada camada e cria, se o número de camadas for muito elevado, um *overhead* de comunicações que não é desprezável;

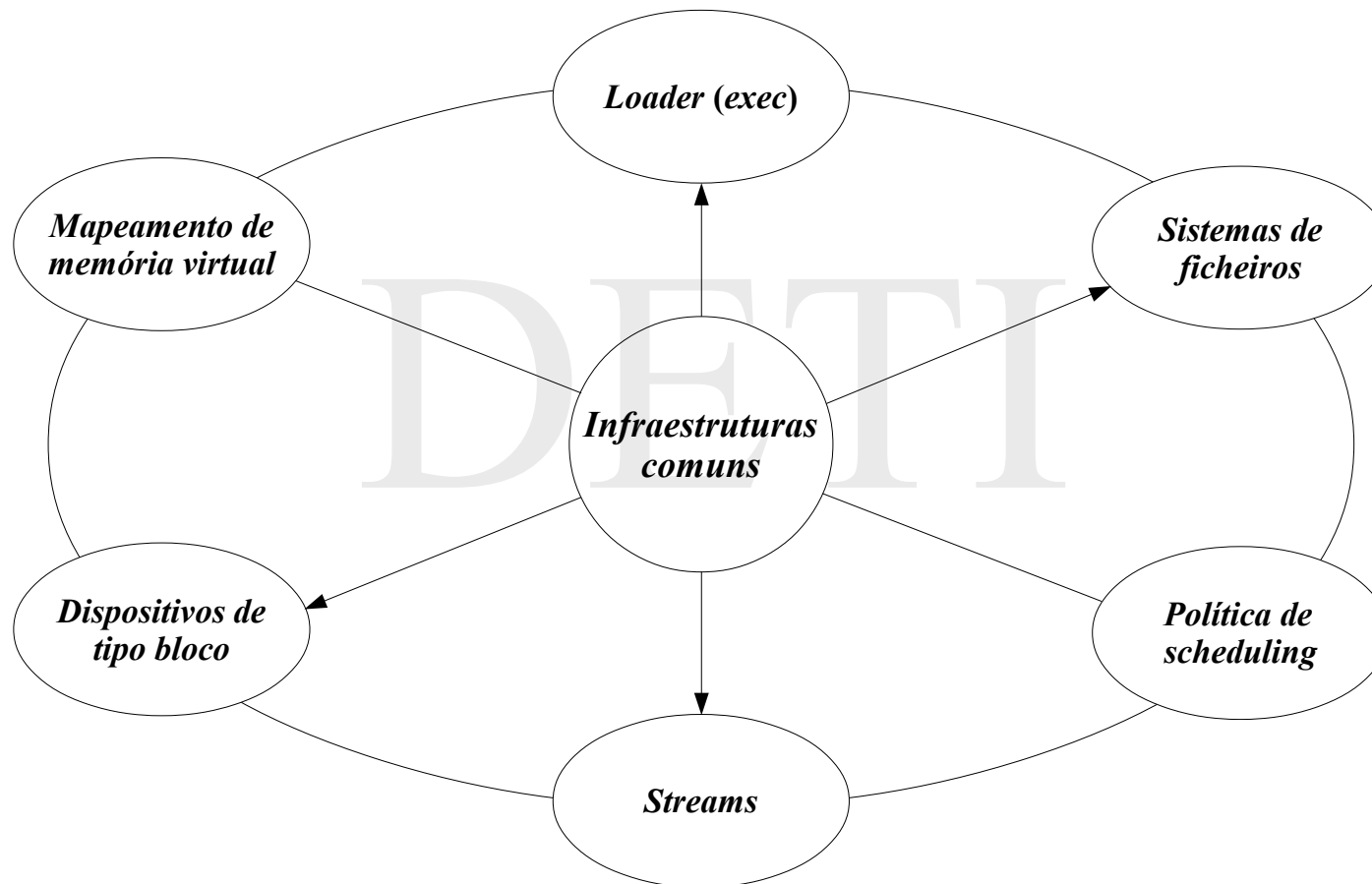
Arquitectura interna - 4

- ♦ *kernel minimalista* – o kernel é reduzido a um número mínimo de funcionalidades, fornecendo tipicamente uma gestão mínima do processador e da memória principal e um mecanismo base de comunicação
- ♦ *microkernel* – as restantes funcionalidades do sistema de operação são implementadas por processos de sistema, executados em modo utilizador, que comunicam entre si e com as aplicações usando o mecanismo de comunicação do kernel
- ♦ *módulos carregados dinamicamente* – as restantes funcionalidades constituem módulos que são carregados e ligados dinamicamente ao kernel com este já em execução; alguns destes módulos, associados a dispositivos *plug-and-play*, por exemplo, podem também ser removidos a qualquer momento.

Arquitectura interna tradicional do Unix



Arquitectura interna contemporânea do Unix



Leituras sugeridas - 1

Operating Systems Concepts, Silberschatz, Galvin, Gagne, John Wiley & Sons, 8th Ed

Capítulo 1: *Introduction*

Conceitos gerais

Capítulo 2: *System Structures*

Elementos e funcionalidades gerais de um sistema de operação

Modern Operating Systems, Tanenbaum, Prentice-Hall International Editions, 3rd Ed

Capítulo 1: *Introduction*

Conceitos gerais

Panorâmica da sua evolução

Aspectos da arquitectura dos sistemas computacionais relevantes à compreensão dos sistemas de operação (muito reduzido)

Elementos e funcionalidades gerais de um sistema de operação

Leituras sugeridas - 2

Operating Systems, W. Stallings, Prentice-Hall International Editions, 7th Ed

Capítulo 1: *Computer System Overview*

Aspectos da arquitectura dos sistemas computacionais relevantes à compreensão dos sistemas de operação (muito abrangente – importante para os alunos que não tiveram aprovação em *AC I* e *AC II*)

Capítulo 2: *Operating System Overview*

Conceitos gerais

Panorâmica da sua evolução

Elementos e funcionalidades gerais de um sistema de operação

Caracterização do Windows e do Unix (Linux)