

VISÃO GERAL

1.1 Introdução

Um *sistema operacional*, por mais complexo que possa parecer, é apenas um conjunto de rotinas executado pelo processador, de forma semelhante aos programas dos usuários. Sua principal função é controlar o funcionamento de um computador, gerenciando a utilização e o compartilhamento dos seus diversos recursos, como processadores, memórias e dispositivos de entrada e saída.

Sem o sistema operacional, um usuário para interagir com o computador deveria conhecer profundamente diversos detalhes sobre hardware do equipamento, o que tornaria seu trabalho lento e com grandes possibilidades de erros. O sistema operacional tem como objetivo funcionar como uma interface entre o usuário e o computador, tornando sua utilização mais simples, rápida e segura.

A grande diferença entre um sistema operacional e aplicações convencionais é a maneira como suas rotinas são executadas em função do tempo. Um sistema operacional não é executado de forma linear como na maioria das aplicações, com início, meio e fim. Suas rotinas são executadas concorrentemente em função de eventos assíncronos, ou seja, eventos que podem ocorrer a qualquer momento.

O nome sistema operacional, apesar de ser o mais empregado atualmente, não é o único para designar esse conjunto de rotinas. Denominações como monitor, executivo, supervisor ou controlador possuem, normalmente, o mesmo significado.

Neste capítulo serão apresentadas as funções básicas de um sistema operacional, o conceito de máquina de camadas, um histórico da evolução dos sistemas operacionais e seus diferentes tipos e classificações.

1.2 Funções Básicas

Um sistema operacional possui inúmeras funções, mas antes de começar o estudo dos conceitos e dos seus principais componentes é importante saber primeiramente quais

são suas funções básicas. Nesta introdução, as funções de um sistema operacional são resumidas em duas, descritas a seguir:

- Facilidade de acesso aos recursos do sistema

Um computador ou sistema computacional possui diversos dispositivos, como monitores de vídeo, impressoras, unidades de CD/DVD, discos e fitas magnéticas. Quando utilizamos um desses dispositivos, não nos preocupamos com a maneira como é realizada esta comunicação e os inúmeros detalhes envolvidos nas operações de leitura e gravação.

Para a maioria dos usuários, uma operação como a leitura de um arquivo em disco pode parecer simples. Na realidade, existe um conjunto de rotinas específicas, controladas pelo sistema operacional, responsável pelo acionamento do mecanismo de leitura e gravação da unidade de disco, posicionamento na trilha e setor corretos, transferência dos dados para a memória e, finalmente, informar ao programa a conclusão da operação. Cabe, então, ao sistema operacional servir de interface entre os usuários e os recursos disponíveis no sistema computacional, tornando esta comunicação transparente, além de permitir um trabalho mais eficiente e com menores chances de erros. Este conceito de ambiente simulado, criado pelo sistema operacional, é denominado máquina virtual e está presente na maioria dos sistemas modernos (Fig. 1.1).

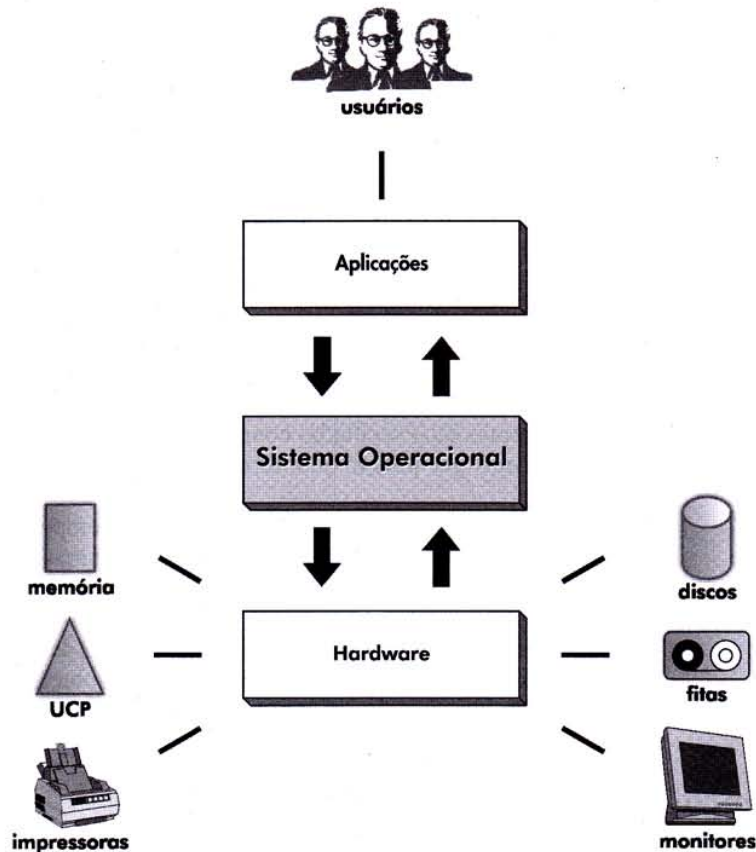


Fig. 1.1 Visão do sistema operacional.

É comum pensar-se que compiladores, linkers, bibliotecas, depuradores e outras ferramentas fazem parte do sistema operacional, mas, na realidade, estes recursos são apenas utilitários, destinados a facilitar a interação do usuário com o computador.

- **Compartilhamento de recursos de forma organizada e protegida**

Em sistemas onde diversos usuários compartilham recursos do sistema computacional, é necessário controlar o uso concorrente desses recursos. Se imaginarmos uma impressora sendo compartilhada, deverá existir algum tipo de controle para que a impressão de um usuário não interfira nas dos demais. Novamente é o sistema operacional que tem a responsabilidade de permitir o acesso concorrente a esse e a outros recursos de forma organizada e protegida.

O compartilhamento de recursos permite, também, a diminuição de custos, na medida em que mais de um usuário pode utilizar as mesmas facilidades concorrentemente, como discos, impressoras, linhas de comunicação etc.

Não é apenas em sistemas multiusuário que o sistema operacional é importante. Se pensarmos que um computador pessoal nos permite executar diversas tarefas ao mesmo tempo, como imprimir um documento, copiar um arquivo pela Internet ou processar uma planilha, o sistema operacional deve ser capaz de controlar a execução concorrente de todas essas atividades.

13 Máquina de Camadas

Um sistema computacional visto somente sob a ótica do *hardware*, ou seja, como um conjunto de circuitos eletrônicos, placas, cabos e fontes de alimentação, tem pouca utilidade. É através do *software* que serviços são oferecidos aos usuários, como armazenamento de dados em discos, impressão de relatórios, geração de gráficos, acesso à Internet, entre outras funções.

Uma operação efetuada pelo software pode ser implementada em hardware, enquanto uma instrução executada pelo hardware pode ser simulada via software. Esta decisão fica a cargo do projetista do sistema computacional em função de aspectos como custo, confiabilidade e desempenho. Tanto o hardware quanto o software são logicamente equivalentes, interagindo de uma forma única para o usuário (Tanenbaum, 1992).

Nos primeiros computadores, a programação era realizada em linguagem de máquina, em painéis através de fios, exigindo, conseqüentemente, um grande conhecimento da arquitetura do hardware. Isso era uma grande dificuldade para os programadores da época (Fig. 1.2a). O surgimento do sistema operacional minimizou esse problema, tornando a interação entre usuário e computador mais simples, confiável e eficiente. A partir desse momento, não existia mais a necessidade de o programador se envolver com a complexidade do hardware para poder trabalhar; ou seja, a parte física do computador tornou-se transparente para o usuário (Fig. 1.2b).

O computador pode ser compreendido como uma *máquina de camadas* ou *máquina de níveis*, onde inicialmente existem dois níveis: hardware (nível 0) e sistema operacional (nível 1). Desta forma, a aplicação do usuário interage diretamente com o sistema operacional, ou seja, como se o hardware não existisse. Esta visão modular e abstrata é chamada de *máquina virtual*.

Na realidade, um computador não possui apenas dois níveis, e sim tantos níveis quantos forem necessários para adequar o usuário às suas diversas aplicações. Quando o usuá-

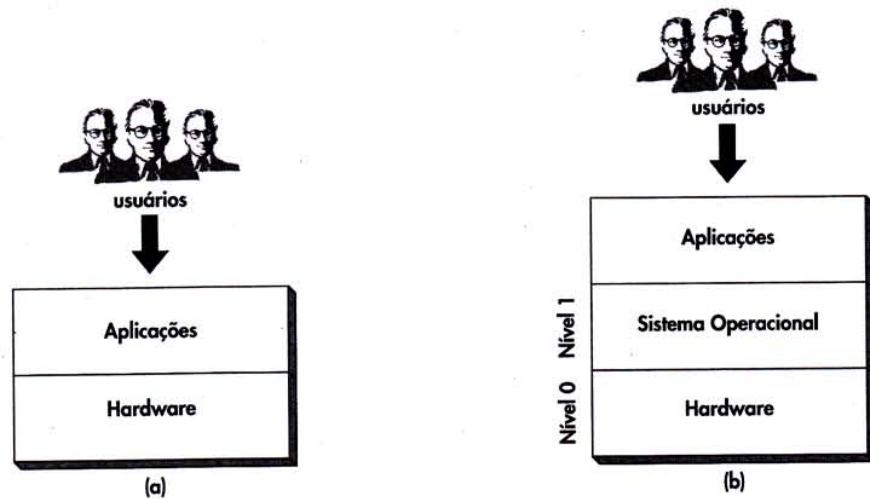


Fig. 1.2 Visão do computador pelo usuário.

rio está trabalhando em um desses níveis não necessita saber da existência das outras camadas, acima ou abaixo de sua máquina virtual.

Atualmente, a maioria dos computadores possui a estrutura básica apresentada na Fig. 1.3, podendo conter um número maior ou menor de camadas. A linguagem utilizada em cada um desses níveis é diferente, variando da mais elementar (baixo nível) à mais sofisticada (alto nível).

1.4 Histórico

A evolução dos sistemas operacionais está, em grande parte, relacionada ao desenvolvimento dos computadores. Neste histórico dividimos essa evolução em décadas, nas

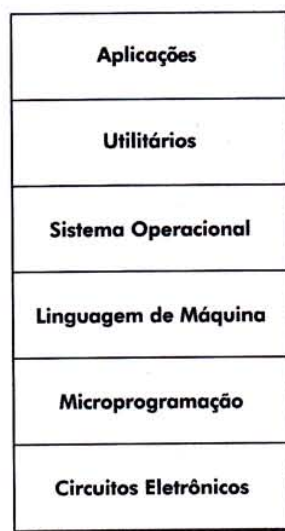


Fig. 1.3 Máquinas de camadas.

quais destacamos em cada uma suas principais características de hardware, software, interação com o sistema e aspectos de conectividade.

Antes da década de 1940, inúmeros esforços foram feitos para criar uma máquina que pudesse realizar cálculos de forma mais rápida e precisa. Em 1642, o matemático francês Blaise Pascal inventou uma máquina de somar para auxiliar seu pai no processo de arrecadação de impostos. Em 1673, o matemático e filósofo alemão Gottfried Leibniz foi além e criou uma máquina capaz de somar e multiplicar, utilizando o conceito de acumulador. Mais tarde, em 1820, o francês Charles Colmar inventaria finalmente uma máquina capaz de executar as quatro operações.

Em 1822 o matemático inglês Charles Babbage criou uma máquina para cálculos de equações polinomiais. Mais tarde, em 1833, Babbage evoluiu esta idéia para uma máquina capaz de executar qualquer tipo de operação, conhecida como Máquina Analítica (*Analytical Engine*). Seu invento é o que mais se aproxima de um computador atual, possuindo os conceitos de unidade central de processamento, memória, unidade de controle e dispositivos de entrada/saída. Enquanto Babbage se preocupava com as características mecânicas do seu invento (hardware), sua discípula Augusta Ada Byron era responsável pela sequência de instruções executadas pela máquina (software). Pelo trabalho realizado na época, Ada Byron é considerada a primeira programadora da história. Devido às limitações técnicas da época, a Máquina Analítica nunca funcionou de forma adequada, mesmo assim Babbage é considerado o “pai do computador”.

Em 1854, o também matemático inglês George Boole criaria a lógica booleana, base para o modelo de computação digital utilizado até hoje. O conceito de lógica binária seria utilizado no desenvolvimento de dispositivos como relés e válvulas, implementados nos primeiros computadores da década de 1940.

No final do século XIX, Herman Hollerith criou um mecanismo utilizando cartões perfurados para acelerar o processamento do censo de 1890 nos EUA. Hollerith fundaria em 1896 a Tabulating Machine Company, que se tornaria a International Business Machine (IBM) em 1924. A utilização de cartões perfurados na computação perduraria por grande parte do século XX, e o nome Hollerith tornar-se-ia sinônimo de cartão perfurado.

Na década de 1930 surgem as primeiras tentativas reais de criar-se uma calculadora eletrônica. Na Alemanha, Konrad Zuse desenvolveu o Z-1, baseado em relés e que utilizava lógica binária. Nos EUA, John Vincent Atanasoff e Clifford Berry desenvolveram uma máquina para o cálculo de equações lineares. Para muitos, o ABC (Atanasoff-Berry Computer) é considerado o primeiro computador eletrônico da história.

Em 1937, o matemático inglês Alan Turing desenvolveu a idéia de Máquina Universal ou Máquina de Turing, capaz de executar qualquer sequência de instruções (algoritmo). Apesar de ser um modelo teórico, a Máquina Universal criou a idéia de “processamento de símbolos”, base da ciência da computação moderna.

14.1 DÉCADA DE 1940

A Segunda Guerra Mundial desencadeou o desenvolvimento de máquinas que pudessem acelerar os procedimentos manuais realizados para fins militares. Neste período surgiram os primeiros computadores eletromecânicos (calculadoras), formados por milhares de válvulas, que ocupavam áreas enormes, sendo de funcionamento lento e duvidoso.

Em 1943, na Inglaterra, Alan Turing estava envolvido no desenvolvimento de uma máquina chamada Colossus para decifrar o código das mensagens alemãs, conhecido como Enigma. Este computador foi desenvolvido e utilizado na quebra de diversos códigos nazistas, como o do "Dia D", e significou uma grande vantagem para os aliados na Segunda Guerra.

Nos EUA, em 1944, foi construído o primeiro computador eletromecânico, batizado de Mark I. Desenvolvido pelo professor Howard Aiken, da Universidade de Harvard, e com apoio da IBM, foi utilizado para cálculos matemáticos pela Marinha. O Mark I utilizava os mesmos princípios da Máquina Analítica, criada por Babbage cem anos antes.

O ENIAC (Electronic Numerical Integrator And Calculator) é considerado o primeiro computador digital e eletrônico. Desenvolvido pelos engenheiros J. Presper Eckert e John W. Mauchly na Universidade da Pensilvânia, foi criado para a realização de cálculos balísticos e, posteriormente, utilizado no projeto da bomba de hidrogênio, ficando em operação no período de 1946 a 1955. Sua estrutura possuía 17 mil válvulas, 10 mil capacitores, 70 mil resistores e pesava 30 toneladas. Quando em operação, consumia cerca de 140 quilowatts e era capaz de realizar 5 mil adições por segundo.

Para trabalhar com o ENIAC era necessário conhecer profundamente o funcionamento do hardware, pois a programação era feita em painéis, através de 6 mil conectores, utilizando linguagem de máquina. Esta tarefa poderia facilmente levar alguns dias. Corretamente programado, um cálculo que levasse vinte e quatro horas manualmente era resolvido em menos de trinta segundos. A diferença entre a velocidade de processamento e o tempo necessário para codificar um programa passou a ser um grande problema a ser resolvido.

O professor John von Neumann, consultor no projeto do ENIAC, imaginou uma máquina de propósito geral na qual tanto instruções quanto dados fossem armazenados em uma mesma memória, tornando o processo de programação muito mais rápido e flexível. Este conceito, aparentemente simples, conhecido como "programa armazenado", é a base da arquitetura de computação atual, batizada de "Arquitetura von Neumann". Finalmente, as mesmas idéias de Babbage (Máquina Analítica) e Turing (Máquina Universal) puderam ser colocadas em prática em um mesmo sistema computacional.

O primeiro computador a implementar o conceito de "programa armazenado" foi o EDSAC (Electronic Delay Storage Automatic Calculator), desenvolvido pelo professor Maurice Wilkes, na Universidade de Cambridge, na Inglaterra, em 1949. Outros computadores foram construídos nessa mesma época com base no mesmo princípio, como o EDVAC (Electronic Discrete Variable Automatic Computer) na Universidade da Pensilvânia, IAS (Institute for Advanced Studies) em Princeton pelo próprio von Neumann, Manchester Mark I, ORDVAC e ELLIAC na Universidade de Illinois, JOHNIAC pela Rand Corp., MANIAC em Los Alamos e WEIZAC em Israel. A maioria destas máquinas foi utilizada apenas em universidades e órgãos militares para cálculos matemáticos.

Nessa fase, os computadores não possuíam ainda dispositivos com função de interface com os usuários, como teclados e monitores, e o conceito de sistema operacional surgiria apenas na década seguinte.

1.4.2 DÉCADA DE 1950

O uso do transistor e da memória magnética contribuiu para o enorme avanço dos computadores da época. O transistor permitiu o aumento da velocidade e da confiabilidade

no processamento, e as memórias magnéticas permitiram o acesso mais rápido aos dados, maior capacidade de armazenamento e computadores menores. Apesar de o seu invento datar do final da década de 1940, os primeiros computadores transistorizados foram lançados comercialmente apenas no final da década de 1950.

Com o desenvolvimento da indústria de computadores, muitas empresas foram criadas ou investiram no setor, como Raytheon, RCA, Burroughs e IBM, o que levou à criação dos primeiros computadores para utilização em aplicações comerciais.

Em 1946, Eckert e Mauchly deixaram a Universidade da Pensilvânia para formar a primeira empresa de computadores, a Eckert-Mauchly Computer Corp. (EMCC), com a intenção de construir o UNIVAC (Universal Automatic Computer). Devido a problemas financeiros, a EMCC foi adquirida pela Remington Rand Corp., possibilitando a conclusão do projeto a tempo de ser utilizado no censo dos EUA em 1951. O UNIVAC I foi o primeiro computador bem-sucedido fabricado para fins comerciais.

Em 1951, o Massachusetts Institute of Technology (MIT) colocou em operação o que é considerado o primeiro computador voltado para o processamento em tempo real, o Whirlwind I. Entre diversas inovações, o Whirlwind introduziu a tecnologia de memória magnética.

Os programas ou jobs passaram a ser perfurados em cartões, que, submetidos a uma leitora, eram gravados em uma fita de entrada (Fig. 1.4a). A fita, então, era lida pelo computador, que executava um programa de cada vez, gravando o resultado do processamento em uma fita de saída (Fig. 1.4b). Ao término de todos os programas, a fita de saída era lida e impressa (Fig. 1.4c). A esse tipo de processamento, em que um conjunto de programas era submetido ao computador, deu-se o nome de *processamento batch*.

Pode não parecer um avanço, mas anteriormente os programas eram submetidos pelo operador, um a um, fazendo com que o processador ficasse ocioso entre a execução de um job e outro. Com o processamento batch, um conjunto de programas era submetido de uma só vez, o que diminuía o tempo de espera entre a execução dos programas, permitindo, assim, melhor aproveitamento do processador.

O primeiro sistema operacional, chamado monitor por sua simplicidade, foi desenvolvido em 1953 pelos usuários do computador IBM 701 do Centro de Pesquisas da General Motors, justamente para tentar automatizar as tarefas manuais até então utilizadas. Posteriormente, este sistema seria reescrito para um computador IBM 704 pelo grupo de usuários da IBM (Weizer, 1981).

Com o surgimento das primeiras linguagens de programação de alto nível, como FORTRAN, ALGOL e COBOL, os programas deixaram de ter relação direta com o hardware dos computadores, o que facilitou e agilizou enormemente o desenvolvimento e a manutenção de programas.

Da mesma maneira que as linguagens de programação, os sistemas operacionais evoluíram no sentido de facilitar o trabalho de codificação, submissão, execução e depuração de programas. Para isso, os sistemas operacionais incorporaram seu próprio conjunto de rotinas para operações de entrada/saída (Input/Output Control System — IOCS). O IOCS eliminou a necessidade de os programadores desenvolverem suas próprias rotinas de leitura/gravação específicas para cada dispositivo. Essa facilidade de comunicação criou o conceito de *independência de dispositivos*, introduzido pelos sistemas operacionais SOS (SHARE Operating System), FMS (FORTRAN Monitor System) e IBSYS, todos para máquinas IBM.

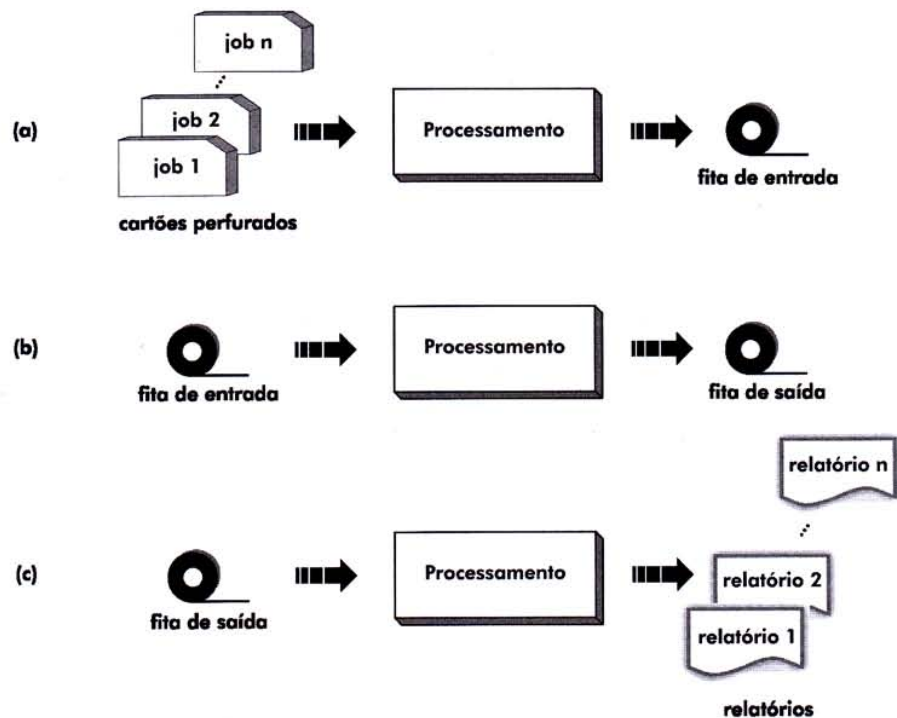


Fig. 1.4 Processamento batch.

No final da década de 1950, a Universidade de Manchester, na Inglaterra, desenvolveu o sistema operacional Atlas, que introduziu a idéia de memória hierarquizada, base do conceito de *memória virtual*, presente hoje na maioria dos sistemas operacionais atuais. O Atlas implementava o esquema de paginação por demanda para transferir informações da memória secundária para a principal.

1.4.3 DÉCADA DE 1960

A partir do surgimento dos circuitos integrados, foi possível viabilizar e difundir o uso de sistemas computacionais nas empresas, devido à redução de seus custos de aquisição. Além disso, houve grande aumento do poder de processamento e diminuição no tamanho dos equipamentos. A década de 1960 foi palco de inúmeras inovações na área de sistemas operacionais, tendo sido implementadas várias técnicas presentes até hoje, como multiprogramação, multiprocessamento, time-sharing e memória virtual.

Uma das características mais importantes surgidas nesta época foi a introdução do conceito de *multiprogramação*. Antes deste conceito, sempre que um programa realizava uma operação de entrada/saída o processador ficava ocioso, aguardando o término da operação. A multiprogramação permitiu que vários programas compartilhassem a memória ao mesmo tempo e, enquanto um programa esperava por uma operação de leitura/gravação, o processador executava um outro programa.

Com a substituição das fitas por discos no processo de submissão dos programas, o processamento batch, juntamente com a multiprogramação, tornou os sistemas mais rápidos e eficientes, pois permitia a carga mais rápida dos programas na memória e a alteração na ordem de execução das tarefas, até então puramente sequencial.

Em 1963, a Burroughs lança o computador B-5000 com o sistema operacional Master Control Program (MCP), que oferecia multiprogramação, memória virtual com segmentação e multiprocessamento assimétrico, além de ser o primeiro sistema a ser desenvolvido em uma linguagem de alto nível. No mesmo ano, a Control Data Corporation anuncia o lançamento do primeiro supercomputador, o CDC 6600, projetado por Seymour Cray.

A IBM lança em 1964 o System/360, que causaria uma revolução na indústria de informática, pois introduzia um conceito de máquinas de portes diferentes, porém com uma mesma arquitetura, permitindo a total compatibilidade entre os diferentes modelos. Desta forma, uma empresa poderia adquirir um modelo mais simples e barato e, conforme suas necessidades, migrar para modelos com mais recursos, sem comprometer suas aplicações já existentes. Para essa série, foi desenvolvido o sistema operacional OS/360.

Os sistemas batch implementando multiprogramação, como o OS/360, foram um grande avanço para os programadores, porém o trabalho de desenvolvimento e depuração das aplicações ainda era lento e tedioso. Caso o programador cometesse apenas um erro de digitação, o tempo de resposta do sistema poderia levar horas. A evolução foi oferecer aos usuários tempos de respostas menores e uma interface que permitisse interagir rapidamente com o sistema. Para tal, cada programa poderia utilizar o processador por pequenos intervalos de tempo. A esse sistema de divisão de tempo chamou-se *tempo compartilhado* (*time-sharing*). Para que a interface entre o computador e usuários fosse possível, foram introduzidos novos dispositivos de entrada/saída, como o terminal de vídeo e o teclado, possibilitando a interação do usuário com a aplicação no decorrer da sua execução (*sistema on-line*).

Um dos primeiros sistemas operacionais de tempo compartilhado foi o CTSS (Compatible Time-Sharing System). Desenvolvido pelo MIT em 1962 para um computador IBM 7094, suportava no máximo 32 usuários interativos, e através de comandos em um terminal permitia compilar e executar seus programas. O CTSS foi a base para outros sistemas operacionais de tempo compartilhado, como o MULTICS (Corbató, 1962).

Em 1965, o MIT, a Bell Labs e a General Electric estavam envolvidos no projeto do sistema operacional MULTICS (Multiplexed Information and Computing Service) para um computador GE 645. Este sistema deveria oferecer vários serviços de forma contínua e confiável, similar aos serviços de luz e telefonia. O MULTICS implementava memória virtual com segmentação e paginação, multiprogramação e deveria suportar múltiplos processadores e usuários. A maior parte do sistema seria desenvolvida em PL/I, uma linguagem de alto nível, para torná-lo portátil, ou seja, independente da plataforma de hardware (Corbató, 1965). Apesar de o MULTICS não ter alcançado seus objetivos, suas idéias influenciariam inúmeros sistemas posteriormente.

A Digital Equipment Corp. (DEC) lançou o PDP-8 em 1965, também revolucionário, pois representava a primeira linha de computadores de pequeno porte e baixo custo, comparativamente aos mainframes até então comercializados, criando o mercado de minicomputadores. Em 1969, Ken Thompson, que trabalhara no projeto do MULTICS, utilizou um PDP-7 para fazer sua própria versão de um sistema operacional que viria a ser conhecido como Unix.

1.4.4 DÉCADA DE 1970

A integração em larga escala (Large Scale Integration — LSI) e a integração em muito larga escala (Very Large Scale Integration — VLSI) levaram adiante o projeto de

miniaturização e barateamento dos equipamentos. Seguindo esta tendência, a Digital lança uma nova linha de minicomputadores, o PDP-11 em 1970 e, posteriormente, o sistema VAX/VMS (Virtual Memory System) de 32 bits.

Em 1971, a Intel Corp. produz seu primeiro microprocessador, o Intel 4004 e, três anos depois, o Intel 8080, utilizado no primeiro microcomputador, o Altair. Posteriormente, a Zilog lançaria um processador concorrente ao da Intel, o Z80. Com a evolução dos microprocessadores, os microcomputadores ganham rapidamente o mercado por serem muito mais baratos que qualquer um dos computadores até então comercializados. Em 1976, Steve Jobs e Steve Wozniak produzem o Apple II de 8 bits, tornando-se um sucesso imediato. Neste mesmo ano, as empresas Apple e a Microsoft são fundadas. O sistema operacional dominante nos primeiros microcomputadores foi o CP/M (Control Program Monitor) da Digital Research.

Para acelerar o processamento foram desenvolvidas arquiteturas com diversos processadores, exigindo dos sistemas operacionais novos mecanismos de controle e sincronismo. O *multiprocessamento* possibilitou a execução de mais de um programa simultaneamente ou até de um mesmo programa por mais de um processador. Além de equipamentos com múltiplos processadores, foram introduzidos processadores vetoriais e técnicas de paralelismo em diferentes níveis, tornando os computadores ainda mais poderosos. Em 1976, o Cray-1 é lançado contendo 200.000 circuitos integrados e realizando 100 milhões de operações de ponto flutuante por segundo (100 MFLOPS).

As redes distribuídas (Wide Area Network — WANs) difundiram-se, permitindo o acesso a outros sistemas de computação, independentemente da distância geográfica. Nesse contexto são desenvolvidos inúmeros protocolos de rede, alguns proprietários, como o DECnet da Digital e o SNA (System Network Architecture) da IBM, e outros de domínio público, como o NCP (predecessor do TCP/IP) e o X.25. Surgem as primeiras redes locais (Local Area Network — LANs) interligando computadores restritos a pequenas áreas. Os sistemas operacionais passam a estar intimamente relacionados aos softwares de rede.

Duas importantes linguagens de programação são desenvolvidas nesta década. Em 1971, o professor Niklaus Wirth desenvolve a linguagem Pascal, voltada para o ensino de técnicas de programação. Em 1975, Dennis Ritchie desenvolve a linguagem C e, juntamente com Ken Thompson, porta o sistema Unix para um PDP-11, concebido inicialmente em assembly.

1.4.5 DÉCADA DE 1980

Em 1981, a IBM entra no mercado de microcomputadores com o IBM PC (Personal Computer), criando a filosofia dos computadores pessoais. O primeiro PC utilizava o processador Intel 8088 de 16 bits e o sistema operacional DOS (Disk Operating System) da Microsoft, muito semelhante ao CP/M.

Na área dos minis e superminicomputadores ganharam impulso os sistemas *multiusuário*, com destaque para os sistemas compatíveis com o Unix. A Universidade de Berkeley, na Califórnia, desenvolveu sua própria versão do sistema Unix (Berkeley Software Distribution – BSD) e introduziu inúmeros melhoramentos, merecendo destaque o protocolo de rede TCP/IP (Transmission Control Protocol/Internet Protocol).

Surgem as estações de trabalho (workstations) que, apesar de serem sistemas monousuário, permitem que sejam executadas diversas tarefas concorrentemente (multitarefa). Em 1982, é fundada a Sun Microsystems, que passaria a atuar fortemen-

te neste setor, lançando as primeiras estações RISC com o sistema operacional SunOS e, posteriormente, Sun Solaris.

Com a evolução dos microprocessadores, principalmente da família Intel, surgem os primeiros sistemas operacionais comerciais que oferecem interface gráfica, como o Microsoft Windows e o OS/2. O software de rede passa a estar fortemente relacionado ao sistema operacional, e surgem os *sistemas operacionais de rede*, com destaque para o Novell Netware e o Microsoft LAN Manager.

Entre os anos de 1982 e 1986, foi desenvolvido no Núcleo de Computação Eletrônica da Universidade Federal do Rio de Janeiro (NCE/UFRJ) o sistema operacional PLURIX para o computador PEGASUS, também construído no NCE. Na década seguinte, o PLURIX seria transportado para a linha de processadores Intel, dando origem ao TROPIX, um sistema operacional multiusuário e multitarefa, de filosofia Unix, disponível gratuitamente na Internet (TROPIX, 2002).

1.4.6 DÉCADA DE 1990

Grandes avanços em termos de hardware, software e telecomunicações foram obtidos nesta década. Essas mudanças foram consequência da evolução das aplicações, que necessitavam cada vez mais de capacidade de processamento e armazenamento de dados, como em sistemas especialistas, sistemas multimídia, banco de dados distribuídos, inteligência artificial e redes neurais.

A evolução da microeletrônica permitiu o desenvolvimento de processadores e memórias cada vez mais velozes e baratos, além de dispositivos de E/S menores, mais rápidos e com maior capacidade de armazenamento. Os componentes baseados em tecnologia VLSI evoluem rapidamente para o ULSI (Ultra Large Scale Integration).

Com o surgimento e a evolução da Internet, o protocolo TCP/IP passou a ser um padrão de mercado, obrigando os fabricantes de sistemas operacionais a oferecer suporte a este protocolo. Devido ao crescimento acelerado da Internet, problemas de gerência, segurança e desempenho tornaram-se fatores importantes relacionados ao sistema operacional e à rede.

A arquitetura cliente/servidor, aplicada basicamente a redes locais, passa a ser utilizada em redes distribuídas como a Internet, permitindo que qualquer pessoa tenha acesso a todo tipo de informação, independentemente de onde esteja armazenada. A partir deste modelo de computação foram criados diversos sistemas dedicados a oferecer serviços, como servidores web, de correio, de arquivos e servidores de banco de dados.

A década de 1990 foi definitiva para a consolidação dos sistemas operacionais baseados em interfaces gráficas. Os conceitos e implementações só vistos em sistemas considerados de grande porte foram introduzidos na maioria dos sistemas para desktop, como na família Windows da Microsoft e no Unix. Em 1991, o finlandês Linus Torvalds começou o desenvolvimento do Linux, que evolui a partir da colaboração de vários programadores que ajudaram no desenvolvimento do kernel, utilitários e vários aplicativos. Atualmente, o Linux é utilizado tanto para fins acadêmicos como comerciais. Em 1993, a Microsoft lança o Windows NT para competir no mercado de servidores corporativos e ao mesmo tempo substituir as versões anteriores do MS-DOS e MS-Windows utilizadas nos computadores pessoais. Durante esta década, o MS-Windows NT e o Unix (HP-UX, IBM-AIX e Sun Solaris) consolidam-se como sistemas para ambientes corporativos.

Outro fato importante nesta década foi o amadurecimento e a popularização do software aberto. Com a evolução da Internet, inúmeros produtos foram desenvolvidos e disponibilizados para uso gratuito, como sistemas operacionais (Linux), banco de dados (MySQL), servidores web (Apache), servidores de correio (Sendmail), dentre outros.

1.4.7 DÉCADA DE 2000

Os computadores da próxima geração devem ser muito mais eficientes que os atuais para atender a demanda cada vez maior de processamento. Para isso, está ocorrendo uma mudança radical na filosofia de projeto de computadores. Arquiteturas paralelas, baseadas em organizações de multiprocessadores não convencionais, já se encontram em desenvolvimento em diversas universidades e centros de pesquisa do mundo.

A forma de interação com os computadores sofrerá, talvez, uma das modificações mais visíveis. Os sistemas operacionais tornam-se cada vez mais intuitivos e simples de serem utilizados. Novas interfaces usuário-máquina serão oferecidas pelos sistemas operacionais, como linguagens naturais, sons e imagens, fazendo essa comunicação mais inteligente, simples e eficiente. A evolução do hardware encadeará modificações profundas nas disciplinas de programação para fazer melhor uso das arquiteturas paralelas.

Os sistemas operacionais passam a ser proativos, ou seja, enquanto no passado o usuário necessitava intervir periodicamente para realizar certas tarefas preventivas e corretivas, os novos sistemas incorporaram mecanismos automáticos de detecção e recuperação de erros. A disponibilidade passa a ser de grande importância para as corporações, e para atender a essa demanda os sistemas em cluster são utilizados em diferentes níveis. Nestes sistemas, computadores são agrupados de forma a oferecer serviços como se fossem um único sistema centralizado. Além de melhorar a disponibilidade, sistemas em cluster permitem aumentar o desempenho e a escalabilidade das aplicações.

O conceito de *processamento distribuído* será explorado nos sistemas operacionais, de forma que suas funções estejam espalhadas por vários processadores através de redes locais e distribuídas. Isso só será possível devido à redução dos custos de comunicação e ao aumento na taxa de transmissão de dados. Com a evolução e consolidação das redes sem fio (wireless), os sistemas operacionais já estão presentes em diversos dispositivos, como em telefones celulares, handhelds e palmtops.

Nesta década, a Microsoft evolui com a linha Windows no sentido de unificar as suas diferentes versões e incluir novos recursos tanto para servidores quanto para computadores pessoais. Os sistemas Windows 2000 e Windows XP, lançados no início da década, evoluíram para o Windows 2003 e o Windows Vista, respectivamente. O Linux evolui para tornar-se o padrão de sistema operacional de baixo custo, com inúmeras formas de distribuições disponíveis no mercado. A comunidade de software livre passa a utilizar o Linux como a base para o desenvolvimento de novas aplicações gratuitas ou de baixo custo.

Os sistemas operacionais desta década caminham para tirar proveito das novas arquiteturas de processadores de 64 bits, especialmente do aumento do espaço de endereçamento, o que permitirá melhorar o desempenho de aplicações que manipulam grandes volumes de dados. Os sistemas Microsoft Windows, o Linux e diversas outras versões do Unix já oferecem suporte aos processadores de 64 bits.

1.5 Tipos de Sistemas Operacionais

Os tipos de sistemas operacionais e sua evolução estão relacionados diretamente com a evolução do hardware e das aplicações por ele suportadas. Muitos termos inicialmente introduzidos para definir conceitos e técnicas foram substituídos por outros, na tentativa de refletir uma nova maneira de interação ou processamento. Isto fica muito claro quando tratamos da unidade de execução do processador. Inicialmente, os termos programa ou job eram os mais utilizados, depois surgiu o conceito de processo e subprocesso e, posteriormente, o conceito de thread.

A seguir, abordaremos os diversos tipos de sistemas operacionais, suas características, vantagens e desvantagens (Fig. 1.5).



Fig. 1.5 Tipos de sistemas operacionais.

1.5.1 SISTEMAS MONOPROGRAMÁVEIS/MONOTAREFA

Os primeiros sistemas operacionais eram tipicamente voltados para a execução de um único programa. Qualquer outra aplicação, para ser executada, deveria aguardar o término do programa corrente. Os *sistemas monoprogramáveis*, como vieram a ser conhecidos, se caracterizam por permitir que o processador, a memória e os periféricos permaneçam exclusivamente dedicados à execução de um único programa.

Os sistemas monoprogramáveis estão tipicamente relacionados ao surgimento dos primeiros computadores na década de 1960. Posteriormente, com a introdução dos computadores pessoais e estações de trabalho na década de 1970, este tipo de sistema voltou a ser utilizado para atender máquinas que, na época, eram utilizadas por apenas um usuário. Os *sistemas monotarefa*, como também são chamados, se caracterizam por permitir que todos os recursos do sistema fiquem exclusivamente dedicados a uma única tarefa.

Neste tipo de sistema, enquanto um programa aguarda por um evento, como a digitação de um dado, o processador permanece ocioso, sem realizar qualquer tipo de processamento. A memória é subutilizada caso o programa não a preencha totalmente, e os periféricos, como discos e impressoras, estão dedicados a um único usuário, nem sempre utilizados de forma integral (Fig. 1.6).

Comparados a outros sistemas, os sistemas monoprogramáveis ou monotarefa são de simples implementação, não existindo muita preocupação com problemas decor-

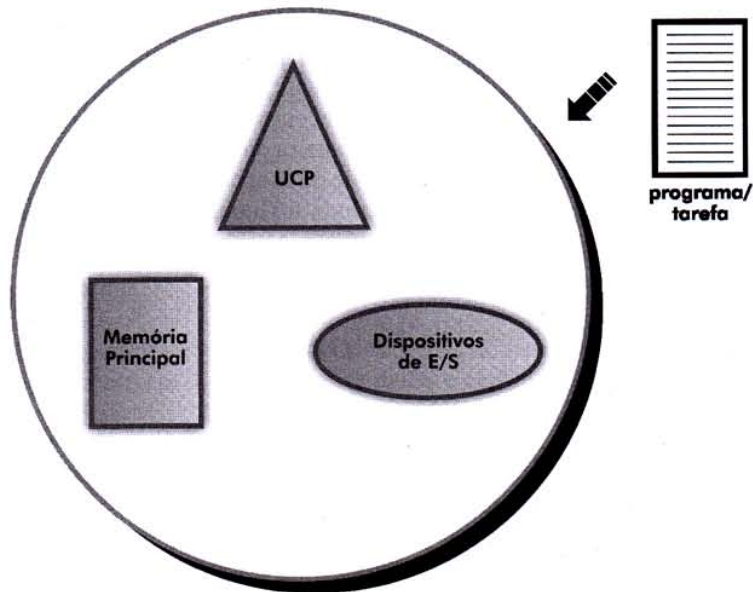


Fig. 1.6 Sistemas monoprogramáveis/monotarefa.

rentes do compartilhamento de recursos, como memória, processador e dispositivos de E/S.

1.5.2 SISTEMAS MULTIPROGRAMÁVEIS/MULTITAREFA

Os *sistemas multiprogramáveis* ou *multitarefa* são uma evolução dos sistemas monoprogramáveis. Neste tipo de sistema, os recursos computacionais são compartilhados entre os diversos usuários e aplicações. Enquanto em sistemas monoprogramáveis existe apenas um programa utilizando os recursos disponíveis, nos multiprogramáveis várias aplicações compartilham esses mesmos recursos.

Neste tipo de sistema, por exemplo, enquanto um programa espera por uma operação de leitura ou gravação em disco, outros programas podem estar sendo processados neste mesmo intervalo de tempo. Nesse caso, podemos observar o compartilhamento da memória e do processador. O sistema operacional se preocupa em gerenciar o acesso concorrente aos seus diversos recursos, como memória, processador e periféricos, de forma ordenada e protegida, entre os diversos programas (Fig. 1.7).

A principal vantagem dos sistemas multiprogramáveis é a redução de custos em função da possibilidade do compartilhamento dos diversos recursos entre as diferentes aplicações. Além disso, sistemas multiprogramáveis possibilitam na média a redução total do tempo de execução das aplicações. Os sistemas multiprogramáveis, apesar de mais eficientes que os monoprogramáveis, são de implementação muito mais complexa.

A partir do número de usuários que interagem com o sistema operacional, podemos classificar os sistemas multiprogramáveis como monousuário ou multiusuário. *Sistemas multiprogramáveis monousuário* são encontrados em computadores pessoais e estações de trabalho, onde há apenas um único usuário interagindo com o sistema. Neste caso existe a possibilidade da execução de diversas tarefas ao mesmo tempo, como a

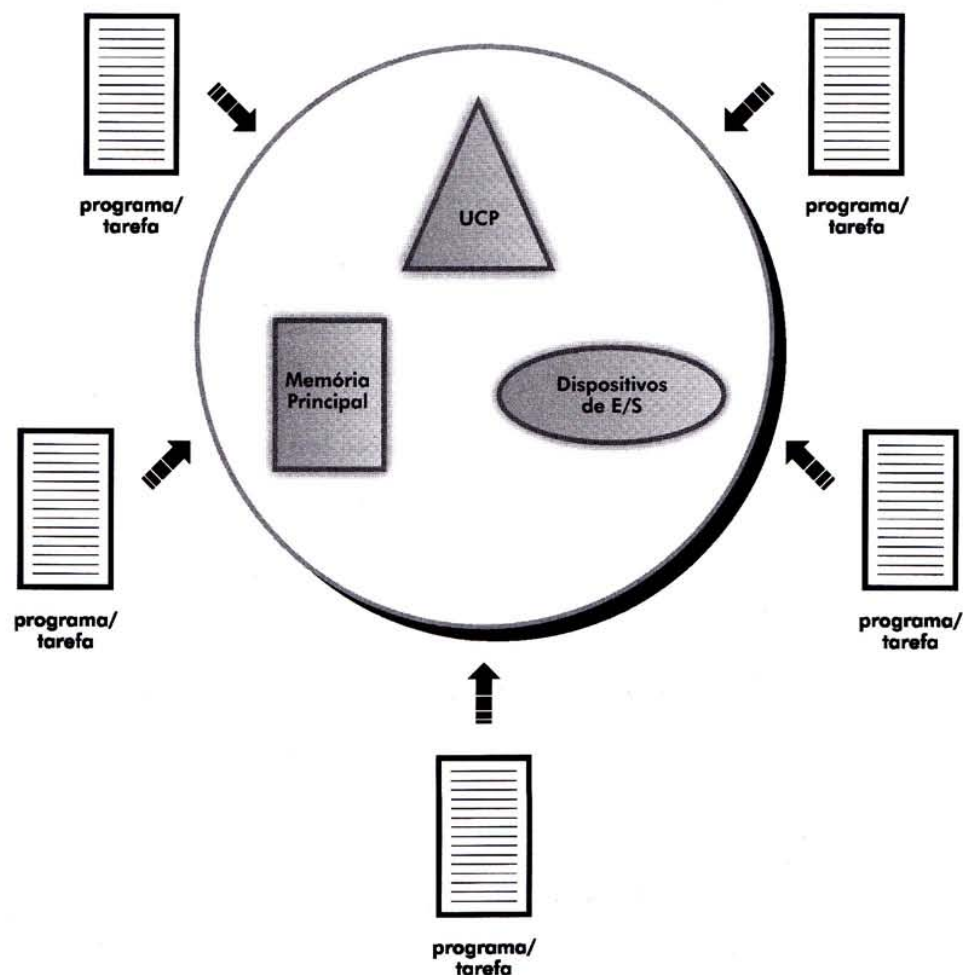


Fig. 1.7 Sistemas multiprogramáveis/multitarefa.

edição de um texto, uma impressão e o acesso à Internet. *Sistemas multiprogramáveis multiusuário* são ambientes interativos que possibilitam a diversos usuários conectarem-se ao sistema simultaneamente. A Tabela 1.1 relaciona os tipos de sistemas em função do número de usuários.

Tabela 1.1 Sistemas × usuários

	Um usuário	Dois ou mais usuários
Monoprogramação/Monotarefa	Monousuário	N/A
Multiprogramação/Multitarefa	Monousuário	Multiusuário

Os sistemas multiprogramáveis ou multitarefa podem ser classificados pela forma com que suas aplicações são gerenciadas, podendo ser divididos em sistemas batch, de tempo compartilhado ou de tempo real. Um sistema operacional pode suportar um ou mais desses tipos de processamento, dependendo de sua implementação (Fig. 1.8).



Fig. 1.8 Tipos de sistemas multiprogramáveis/multitarefa.

1.5.2.1 Sistemas batch

Os *sistemas batch* foram os primeiros tipos de sistemas operacionais multiprogramáveis a serem implementados na década de 1960. Os programas, também chamados de *jobs*, eram submetidos para execução através de cartões perfurados e armazenados em disco ou fita, onde aguardavam para ser processados. Posteriormente, em função da disponibilidade de espaço na memória principal, os jobs eram executados, produzindo uma saída em disco ou fita.

O *processamento batch* tem a característica de não exigir a interação do usuário com a aplicação. Todas as entradas e saídas de dados da aplicação são implementadas por algum tipo de memória secundária, geralmente arquivos em disco. Alguns exemplos de aplicações originalmente processadas em batch são programas envolvendo cálculos numéricos, compilações, ordenações, backups e todos aqueles onde não é necessária a interação com o usuário.

Esses sistemas, quando bem projetados, podem ser bastante eficientes, devido à melhor utilização do processador, entretanto podem oferecer tempos de resposta longos. Atualmente, os sistemas operacionais implementam ou simulam o processamento batch, não existindo sistemas exclusivamente dedicados a este tipo de processamento.

1.5.2.2 Sistemas de tempo compartilhado

Os *sistemas de tempo compartilhado* (*time-sharing*) permitem que diversos programas sejam executados a partir da divisão do tempo do processador em pequenos intervalos, denominados *fatia de tempo* (*time-slice*). Caso a fatia de tempo não seja suficiente para a conclusão do programa, ele é interrompido pelo sistema operacional e substituído por um outro, enquanto fica aguardando por uma nova fatia de tempo. O sistema cria para cada usuário um ambiente de trabalho próprio, dando a impressão de que todo o sistema está dedicado exclusivamente a ele.

Geralmente, sistemas de tempo compartilhado permitem a interação dos usuários com o sistema através de terminais que incluem vídeo, teclado e mouse. Esses sistemas possuem uma linguagem de controle que permite ao usuário comunicar-se diretamente com o sistema operacional através de comandos. Desta forma é possível verificar arquivos armazenados em disco ou cancelar a execução de um programa. O sistema, normalmente, responde em poucos segundos à maioria desses comandos. Devido a esse tipo de interação, os sistemas de tempo compartilhado também ficaram conhecidos como *sistemas on-line*.

A maioria das aplicações comerciais atualmente é processada em sistemas de tempo compartilhado, pois elas oferecem tempos de respostas razoáveis a seus usuários e custos mais baixos, em função da utilização compartilhada dos diversos recursos do sistema.

1.5.2.3 Sistemas de tempo real

Os *sistemas de tempo real* (*real-time*) são implementados de forma semelhante aos sistemas de tempo compartilhado. O que caracteriza a diferença entre os dois tipos de sistemas é o tempo exigido no processamento das aplicações. Enquanto em sistemas de tempo compartilhado o tempo de processamento pode variar sem comprometer as aplicações em execução, nos sistemas de tempo real os tempos de processamento devem estar dentro de limites rígidos, que devem ser obedecidos, caso contrário poderão ocorrer problemas irreparáveis.

Nos sistemas de tempo real não existe a idéia de fatia de tempo, implementada nos sistemas de tempo compartilhado. Um programa utiliza o processador o tempo que for necessário ou até que apareça outro mais prioritário. A importância ou prioridade de execução de um programa é definida pela própria aplicação e não pelo sistema operacional.

Esses sistemas, normalmente, estão presentes em aplicações de controle de processos, como no monitoramento de refinarias de petróleo, controle de tráfego aéreo, de usinas termoeletricas e nucleares, ou em qualquer aplicação onde o tempo de processamento é fator fundamental.

1.5.3 SISTEMAS COM MÚLTIPLOS PROCESSADORES

Os *sistemas com múltiplos processadores* caracterizam-se por possuir duas ou mais UCPs interligadas e trabalhando em conjunto. A vantagem deste tipo de sistema é permitir que vários programas sejam executados ao mesmo tempo ou que um mesmo programa seja subdividido em partes para serem executadas simultaneamente em mais de um processador.

Com múltiplos processadores foi possível a criação de sistemas computacionais voltados, principalmente, para processamento científico, aplicado, por exemplo, no desenvolvimento aeroespacial, prospecção de petróleo, simulações, processamento de imagens e CAD. A princípio qualquer aplicação que faça uso intensivo da UCP será beneficiada pelo acréscimo de processadores ao sistema. A evolução desses sistemas deve-se, em grande parte, ao elevado custo de desenvolvimento de processadores de alto desempenho.

Os conceitos aplicados ao projeto de sistemas com múltiplos processadores incorporam os mesmos princípios básicos e benefícios apresentados na multiprogramação, além de outras características e vantagens específicas como escalabilidade, disponibilidade e balanceamento de carga.

Escalabilidade é a capacidade de ampliar o poder computacional do sistema apenas adicionando novos processadores. Em ambientes com um único processador, caso haja problemas de desempenho, seria necessário substituir todo o sistema por uma outra configuração com maior poder de processamento. Com a possibilidade de múltiplos processadores, basta acrescentar novos processadores à configuração.

Disponibilidade é a capacidade de manter o sistema em operação mesmo em casos de falhas. Neste caso, se um dos processadores falhar, os demais podem assumir suas



Fig. 1.9 Tipos de sistemas com múltiplos processadores.

funções de maneira transparente aos usuários e suas aplicações, embora com menor capacidade de computação.

Balanceamento de carga é a possibilidade de distribuir o processamento entre os diversos processadores da configuração a partir da carga de trabalho de cada processador, melhorando, assim, o desempenho do sistema como um todo.

Um fator-chave no desenvolvimento de sistemas operacionais com múltiplos processadores é a forma de comunicação entre as UCPs e o grau de compartilhamento da memória e dos dispositivos de entrada e saída. Em função desses fatores, podemos classificar os sistemas com múltiplos processadores em fortemente acoplados ou fracamente acoplados (Fig. 1.9).

A grande diferença entre os dois tipos de sistemas é que em sistemas fortemente acoplados existe apenas uma memória principal sendo compartilhada por todos os processadores, enquanto nos fracamente acoplados cada sistema tem sua própria memória individual. Além disso, a taxa de transferência entre processadores e memória em sistemas fortemente acoplados é muito maior que nos fracamente acoplados.

1.5.3.1 Sistemas fortemente acoplados

Nos *sistemas fortemente acoplados* (tightly coupled) existem vários processadores compartilhando uma única memória física (shared memory) e dispositivos de entrada/saída sendo gerenciados por apenas um sistema operacional (Fig. 1.10). Em função destas características, os sistemas fortemente acoplados também são conhecidos como *multiprocessadores*.

Os sistemas fortemente acoplados podem ser divididos em SMP (Symmetric Multiprocessors) e NUMA (Non-Uniform Memory Access). Os *sistemas SMP* caracterizam-se pelo tempo uniforme de acesso à memória principal pelos diversos processadores. Os *sistemas NUMA* apresentam diversos conjuntos reunindo processadores e memória principal, sendo que cada conjunto é conectado aos outros através de uma rede de interconexão. O tempo de acesso à memória pelos processadores varia em função da sua localização física.

Nos *sistemas SMP* e *NUMA* todos os processadores têm as mesmas funções. Inicialmente, os sistemas com múltiplos processadores estavam limitados aos sistemas de grande porte, restritos ao ambiente universitário e às grandes corporações. Com a evolução dos computadores pessoais e das estações de trabalho, os sistemas multitarefa

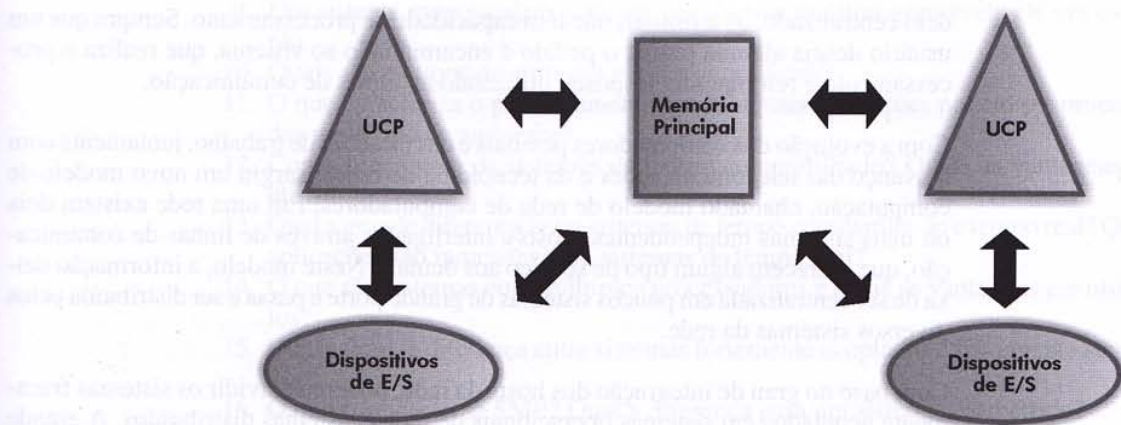


Fig. 1.10 Sistemas fortemente acoplados.

evoluíram para permitir a existência de vários processadores no modelo simétrico. Atualmente, a grande maioria dos sistemas operacionais, como o Unix e o Microsoft Windows, implementa esta funcionalidade.

1.5.3.2 Sistemas fracamente acoplados

Os *sistemas fracamente acoplados* (loosely coupled) caracterizam-se por possuir dois ou mais sistemas computacionais conectados através de linhas de comunicação. Cada sistema funciona de forma independente, possuindo seu próprio sistema operacional e gerenciando seus próprios recursos, como UCP, memória e dispositivos de entrada/saída (Fig. 1.11). Em função destas características, os sistemas fracamente acoplados também são conhecidos como *multicomputadores*. Neste modelo, cada sistema computacional também pode ser formado por um ou mais processadores.

Até meados da década de 1980, as aplicações eram tipicamente centralizadas em sistemas de grande porte, com um ou mais processadores. Neste tipo de configuração, os usuários utilizam terminais não inteligentes conectados a linhas seriais dedicadas ou linhas telefônicas públicas para a comunicação interativa com esses sistemas. No mo-

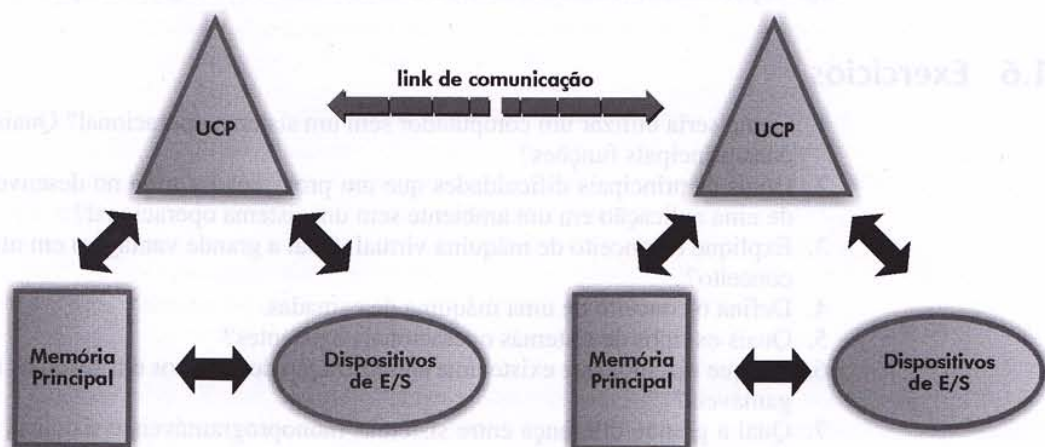


Fig. 1.11 Sistemas fracamente acoplados.

delo centralizado, os terminais não têm capacidade de processamento. Sempre que um usuário deseja alguma tarefa, o pedido é encaminhado ao sistema, que realiza o processamento e retorna uma resposta, utilizando as linhas de comunicação.

Com a evolução dos computadores pessoais e das estações de trabalho, juntamente com o avanço das telecomunicações e da tecnologia de redes, surgiu um novo modelo de computação, chamado modelo de rede de computadores. Em uma rede existem dois ou mais sistemas independentes (hosts), interligados através de linhas de comunicação, que oferecem algum tipo de serviço aos demais. Neste modelo, a informação deixa de ser centralizada em poucos sistemas de grande porte e passa a ser distribuída pelos diversos sistemas da rede.

Com base no grau de integração dos hosts da rede, podemos dividir os sistemas fracamente acoplados em sistemas operacionais de rede e sistemas distribuídos. A grande diferença entre os dois modelos é a capacidade do sistema operacional em criar uma imagem única dos serviços disponibilizados pela rede.

Os *sistemas operacionais de rede* (SOR) permitem que um host compartilhe seus recursos, como uma impressora ou diretório, com os demais hosts da rede. Um exemplo deste tipo de sistema são as redes locais, onde uma estação pode oferecer serviços de arquivos e impressão para as demais estações da rede, dentre outros serviços.

Enquanto nos SORs os usuários têm o conhecimento dos hosts e seus serviços, nos *sistemas distribuídos* o sistema operacional esconde os detalhes dos hosts individuais e passa a tratá-los como um conjunto único, como se fosse um sistema fortemente acoplado. Os sistemas distribuídos permitem, por exemplo, que uma aplicação seja dividida em partes e que cada parte seja executada por hosts diferentes da rede de computadores. Para o usuário e suas aplicações é como se não existisse a rede de computadores, mas sim um único sistema centralizado.

Outro exemplo de sistema distribuído são os *clusters*. Em um cluster existem dois ou mais servidores ligados, normalmente, por algum tipo de conexão de alto desempenho. O usuário não conhece os nomes dos membros do cluster e não sabe quantos são. Quando ele precisa de algum serviço, basta solicitar ao cluster para obtê-lo. Atualmente, sistemas em cluster são utilizados para serviços de banco de dados e Web, garantindo alta disponibilidade, escalabilidade e balanceamento de carga à solução.

1.6 Exercícios

1. Como seria utilizar um computador sem um sistema operacional? Quais são suas duas principais funções?
2. Quais as principais dificuldades que um programador teria no desenvolvimento de uma aplicação em um ambiente sem um sistema operacional?
3. Explique o conceito de máquina virtual. Qual a grande vantagem em utilizar este conceito?
4. Defina o conceito de uma máquina de camadas.
5. Quais os tipos de sistemas operacionais existentes?
6. Por que dizemos que existe uma subutilização de recursos em sistemas monoprogramáveis?
7. Qual a grande diferença entre sistemas monoprogramáveis e sistemas multiprogramáveis?
8. Quais as vantagens dos sistemas multiprogramáveis?

9. Um sistema monousuário pode ser um sistema multiprogramável? Dê um exemplo.
10. Quais são os tipos de sistemas multiprogramáveis?
11. O que caracteriza o processamento batch? Quais aplicações podem ser processadas neste tipo de ambiente?
12. Como funcionam os sistemas de tempo compartilhado? Quais as vantagens em utilizá-los?
13. Qual a grande diferença entre sistemas de tempo compartilhado e tempo real? Quais aplicações são indicadas para sistemas de tempo real?
14. O que são sistemas com múltiplos processadores e quais as vantagens em utilizá-los?
15. Qual a grande diferença entre sistemas fortemente acoplados e fracamente acoplados?
16. O que é um sistema SMP? Qual a diferença para um sistema assimétrico?
17. O que é um sistema fracamente acoplado? Qual a diferença entre sistemas operacionais de rede e sistemas operacionais distribuídos?
18. Quais os benefícios de um sistema com múltiplos processadores em um computador pessoal?
19. Qual seria o tipo de sistema operacional recomendável para uso como servidor de aplicações em um ambiente corporativo?
20. Qual seria o tipo de sistema operacional recomendável para executar uma aplicação que manipula grande volume de dados e necessita de um baixo tempo de processamento?