

Capítulo 1

Introdução aos Sistemas Operacionais

Um sistema operacional é um programa que atua como uma *interface* entre o *hardware* do computador e o usuário do sistema. Seu propósito é fornecer um ambiente no qual se possa executar programas. Suas metas são tornar o sistema do computador conveniente ao uso e que a utilização do *hardware* seja feita de modo eficiente.

Neste capítulo são apresentados uma definição do que é um sistema operacional, seus objetivos e divisão por áreas de aplicação. A seguir, é feito um breve histórico do desenvolvimento dos sistemas operacionais.

1. Introdução aos Sistemas Operacionais

1.1. Definição de Sistemas Computacionais

Um **sistema computacional** é um conjunto de recursos computacionais, sendo uma parte implementada em *hardware* e outra parte em *software*, dependendo do projeto, para interagirem cooperativamente.

Em torno de um sistema computacional, existem usuários com problemas distintos para serem resolvidos, por exemplo, um usuário precisa editar texto, enquanto outro precisa fazer a contabilidade da empresa. Deste modo, o problema de cada usuário deve ser atendido por um programa, ou aplicativo, específico que, neste caso, seriam, respectivamente, um editor de textos e um programa de contabilidade. Além disso, o sistema computacional deve possuir o *hardware* necessário, ou seja, os dispositivos físicos ou recursos físicos, para a execução desses programas.

De um modo geral, programas possuem muito em comum, por exemplo, tanto editor de texto como o sistema de contabilidade precisam acessar o disco, e a forma de acesso aos periféricos é a mesma para todos os programas.

Além disso, para um melhor aproveitamento dos recursos computacionais, os sistemas computacionais são projetados para que vários usuários possam compartilhar os recursos simultaneamente. Neste caso, os aplicativos podem apresentar necessidades conflitantes, pois muitas vezes disputam os mesmos recursos. Por exemplo, o editor de texto e o sistema de contabilidade podem solicitar, ao mesmo tempo, a única impressora do sistema computacional.

Assim, qualquer pessoa que utilize um sistema computacional deve saber que existe um software chamado **sistema operacional** (ao longo do curso, será referenciado muitas vezes apenas como S.O.), que de alguma forma controla os recursos do sistema computacional, qualquer que seja o seu tipo ou porte. Sistemas operacionais são necessários nos mais simples microcomputadores domésticos ou nos mais sofisticados supercomputadores de institutos de pesquisa ou de grandes corporações.

1.2. Objetivos da Disciplina

O correto entendimento dos mecanismos presentes nos S.O. permite ao profissional de informática uma melhor compreensão do seu ambiente de trabalho, resultando no desenvolvimento de soluções com maior qualidade e eficiência. Assim, os objetivos da disciplina são permitir ao profissional de informática a atuar nas áreas de suporte, seleção e recomendação de S.O., a obter visão básica do desenvolvimento e implementação de um S.O. e, principalmente, fornecer subsídios necessários para a compreensão do funcionamento de S.O.

1.3. Definição do Sistema Operacional

O sistema operacional é um programa que controla e coordena o uso do *hardware* do computador entre os vários programas de aplicação para os vários usuários. Assim, podemos dizer que o sistema operacional é um conjunto de módulos de *software* que regem os recursos do sistema, resolvem seus conflitos, simplificam o uso da máquina e otimizam seu desempenho global.

Um sistema computacional é constituído de vários elementos, que atuam em diferentes níveis, como exemplificado na figura 1.1. Observe que o sistema operacional, juntamente com aplicações que constituem o software do sistema, pertence a uma mesma camada, cujo objetivo fundamental é atender às solicitações dos programas de aplicação do usuário por funcionalidades desempenhadas pelo *hardware* do sistema.

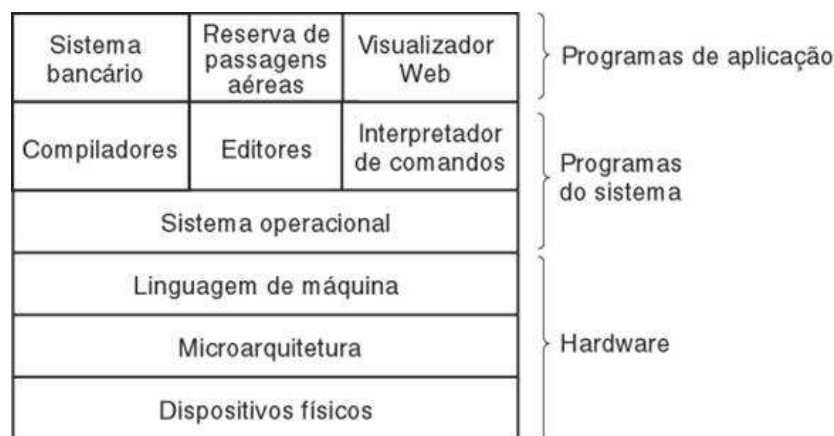


Figura 1.1. Estrutura simplificada de um sistema computacional

Um sistema operacional deve ter o completo domínio sobre os recursos da máquina. O escalonamento de recursos, o controle de entrada e saída (E/S), a gerência da memória, a gerência do processador, o escalonamento de processos e a segurança são funções que o sistema operacional deve exercer.

No projeto de um sistema operacional deve ter em mente dois objetivos principais:

- Apresentar ao usuário do computador uma forma amena de utilizar a máquina. Criar uma máquina virtual, de fácil compreensão para o usuário, com características diferentes da máquina física;
- Realizar o melhor uso possível do *hardware* disponível, aumentando o desempenho do sistema e diminuindo o custo.

Finalmente, o sistema operacional pode ser visto como uma **máquina estendida** (ou máquina virtual), ocultando detalhes complexos do funcionamento do hardware que constitui o sistema computacional, e como um **gerenciador de recursos**, que podem ser compartilhados no **tempo** (diferentes programas ou usuários aguardam sua vez de usá-los) ou no **espaço** (vários programas ou usuários utilizam simultaneamente uma parte do recurso), por exemplos, várias aplicações ocupam partes da memória do sistema.

Como sistemas computacionais são empregados em ambientes de tarefas bastante heterogêneos, é razoável assumir que existam diferentes sistemas operacionais (ou diferentes versões de um mesmo), que sejam mais adequadas para a realização de um conjunto de tarefas específico. É possível dividir as **áreas de aplicação** de um sistema operacional em **sistemas de tempo real** e **sistemas de processamento de tarefas**, definidos a seguir.

1.4. Classificação de Sistemas de Computação em Relação ao Funcionamento

Os diversos tipos de sistemas computacionais diferem entre si, principalmente nas tarefas de administração dos processos e capacidade de atendimento aos processos, por exemplo, alguns sistemas computacionais tratam um só processo de usuário por vez, ou seja, mantêm apenas um processo de usuário na memória, enquanto outros sistemas computacionais já suportam múltiplos processos de simultaneamente em memória.

Entretanto, o simples fato de existir mais de um processo em memória, não garante que todos, ou alguns, serão atendidos ao mesmo tempo, ou seja, paralelamente. A estrutura de funcionamento de um sistema computacional depende dos recursos de *hardware* presentes, bem como da capacidade do sistema operacional em atender as solicitações dos processos, em disponibilizar os recursos aos processos e ao escalonamento de uso do(s) processador(es).

A seguir veremos uma classificação de sistemas computacionais, considerando a estrutura de funcionamento dos mesmos, porém lembramos que devemos complementar esse estudo em outros textos, pois existem diferenças de classificação entre diferentes autores. Assim, o que apresentamos a seguir são conceitos básicos, sem seguir especificamente um único autor.

a) Sistemas monoprogramados:

São os mais simples e foram largamente utilizados. Permitiam que somente um programa por vez fosse processado, ou seja, somente um programa tinha posse de todos os recursos do sistema computacional, indiferente da sua necessidade de uso, conseqüentemente, a utilização dos recursos não será otimizada. Além disso, o programa em execução deverá administrar todo o sistema computacional. Se este tipo de controle for utilizado em equipamentos caros, a relação custo x benefício tende a ser desvantajosa.

b) Sistemas multiprogramados:

Com o objetivo de melhorar a relação custo x benefício e visando a otimização do uso dos recursos, surge, no início dos anos 70, o conceito e a primeira implementação de multiprogramação. A idéia era otimizar a utilização de recursos destinando determinadas tarefas a componentes específicos do sistema computacional, por exemplo, quando executa-se uma rotina de entrada e saída (E/S ou I/O), pode-se atribuir o procedimento de E/S a um hardware/software dedicado à função. Assim, o processador ficaria ocioso, podendo ser utilizado por outro processo.

Para tanto, deve ser mantido em memória, simultaneamente, uma quantidade n de processos, capazes de ocupar o processador durante todo o tempo. Uma variação um pouco mais sofisticada dos sistemas multiprogramados, são os sistemas que fazem multitarefa cooperativa, que seguem o mesmo conceito com algumas melhorias.

c) Sistemas de tempo compartilhado:

Os sistemas de tempo compartilhado (*time sharing* ou *time shared*), da mesma forma que os sistemas multiprogramados, possibilitavam que, em um instante de tempo vários processos fossem alocados na memória, entretanto, a execução se daria em intervalos cíclicos de tempo, determinando uma “fatia” de tempo (*time slice*) para cada processo.

O tempo compartilhado é largamente utilizado nos equipamentos de grande porte e permitiu a implementação de sistemas multiusuários, onde vários usuários, de forma simultânea e *online*, concorrem pelo tempo de execução. Como exemplo de sistemas de tempo compartilhado, podemos citar os sistemas de multitarefa preemptiva.

d) Sistemas multiexecutados:

Outra evolução “natural” dos sistemas monoprogramados são os sistemas multiexecutados, pois a própria evolução do hardware permitiu a execução de várias tarefas concorrentemente, através de tempo compartilhado, sendo que essas tarefas são comandadas por um único usuário (monousuário). Ou seja, os sistemas multiexecutados são similares aos de tempo compartilhado, porém não permitem multiusuários.

e) Sistemas multiprocessados:

Os sistemas multiprocessados permitem que durante um determinado instante de tempo, vários processos estejam alocados na memória, sendo executados simultaneamente (processamento paralelo), não de forma concorrente, como em outros sistemas multitarefas (por exemplo, sistemas exclusivamente multiprogramado ou exclusivamente por tempo compartilhado). Os sistemas computacionais que suportam multiprocessamento possuem mais do que um elemento processador.

A execução de dos programas nos diversos processadores não impede a utilização dos periféricos em outras tarefas independente dos processadores (assim como, nos sistemas de tempo compartilhado, multiprogramados e multi-executados) de forma a otimizar o uso destes. Sistemas multiprocessados são ambientes com sistemas operacionais complexos e sofisticados destinados a sistemas computacionais de alto desempenho.

1.5. Conceitos Básicos de Sistemas Operacionais

Um sistema computacional é composto por um ou mais processadores, uma certa quantidade de memória, terminais, discos magnéticos, interfaces de rede e dispositivos de E/S, etc., ou seja, estamos lidando com um sistema extremamente

complexo, e desenvolver software que gerencie e integre esses componentes, fazendo-os trabalhar corretamente e de forma otimizada, não é tarefa fácil.

O sistema operacional é uma camada de software colocada entre os recursos de hardware e os programas que executam tarefas para os usuários, sendo que uma de suas principais responsabilidades é permitir e promover o acesso aos periféricos, sempre que um programa solicitar uma operação de E/S. Através dessa intervenção do sistema operacional, o programador não precisa conhecer detalhes do hardware, por exemplo, informações de como “enviar um caractere para a impressora”, são internas ao sistema operacional. Além disso, como todos os acessos aos periféricos são feitos através do sistema operacional, fica mais fácil controlar a disponibilização dos recursos, buscando uma distribuição justa, eficiente e conveniente, não só dos recursos de E/S, mas de todo o sistema computacional.

Uma utilização mais eficiente do sistema computacional é obtida a partir da distribuição de seus recursos entre os programas, por exemplo, a distribuição do espaço em disco, da memória principal, do tempo do processador, do acesso a disco, etc. Já a utilização mais conveniente é obtida a partir da disponibilização dos recursos do sistema computacional, sem que os usuários conheçam os detalhes dos recursos.

Como exemplo, vamos imaginar um usuário especializado, um programador, que ao desenvolver um programa precisa colocar um caractere na tela. Para tanto, em geral, é necessária toda uma sequência de acessos à interface do vídeo, diversos registradores devem ser lidos ou escritos e, além disso, podem existir diferentes tipos de interfaces que exigirão diferentes sequências de acesso. Porém, através do sistema operacional, o programador apenas informa, no programa, qual caractere deve ser colocado na tela e todo o trabalho de acesso ao periférico é feito pelo sistema operacional.

Ao esconder detalhes dos periféricos, muitas vezes são criados recursos de mais alto nível, ou seja, as abstrações. Por exemplo, os programas utilizam o espaço em disco através do conceito de arquivo. Arquivos não existem no hardware, mas formam um recurso criado a partir do que o hardware oferece. Para o programador é muito mais confortável trabalhar com arquivos do que receber uma área de espaço em disco que ele próprio teria que organizar.

Como exemplo de abstração, considere uma instrução de E/S, por exemplo, READ ou WRITE em um IBM PC, que deve ser acompanhada de 13 parâmetros, especificando o endereço do bloco a ser lido, o número de setores por trilha, o modo de gravação no meio físico, o espaçamento entre setores, etc., sem contar com as tarefas de ordem física/mecânica, por exemplo, verificar se o motor do drive já está acionado. A grande maioria dos programadores não se envolve com tais detalhes, pois lida com uma abstração de alto nível e, portanto mais simples. No exemplo em questão, a abstração feita nos discos é visualizá-los como uma coleção de arquivos identificados por nomes, onde os eventos, a manipulação dos arquivos, não consideram maiores detalhes e restringem-se a simplesmente abrir, ler/escrever e fechar.

1.6. Arquitetura de Sistemas Operacionais

A arquitetura de um sistema operacional é a estrutura básica sobre a qual é projetado o sistema operacional, de como as abstrações são realmente implementadas, como o sistema computacional deve ser solicitado e atender aos aplicativos, como interagem as partes do sistema operacional entre si e como o sistema operacional responde às solicitações dos aplicativos.

1.6.1. Arquitetura monolítica:

É a arquitetura mais antiga e mais comum. Cada componente do sistema operacional é contido no núcleo (*kernel*) e pode comunicar-se com qualquer outro componente diretamente. Essa intercomunicação direta permite rapidez na resposta de sistema operacional monolíticos, entretanto como núcleos monolíticos agrupam os componentes todos juntos, é difícil identificar a origem de um determinado problema ou erro. Além disso, todo o código do sistema operacional é executado com acesso irrestrito ao sistema, o que pode facilitar a ocorrência de danos provocados intencionalmente, ou não, por outros aplicativos.

1.6.2. Arquitetura em camadas:

À medida que os sistemas operacionais tornaram-se mais complexos e maiores, projetos puramente monolíticos tornaram-se inviáveis e, então a arquitetura em camada, ou modular, tornou-se uma boa opção, agrupando “camadas” de componentes, ou seja, conjunto de procedimentos, que realizam tarefas similares.

Cada camada comunica-se somente com as suas camadas imediatamente inferior e superior. Uma camada inferior sempre presta um serviço à sua camada superior, sendo que a camada superior não sabe como o serviço é feito, apenas o solicita. A implementação de uma camada pode ser modificada sem exigir modificação em outra camada, pois possuem componentes autocontidos.

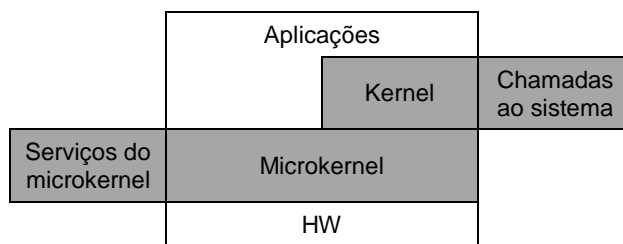
Em uma abordagem em camadas, a solicitação de um serviço pode precisar passar por muitas camadas antes de ser atendida, assim o desempenho se degrada em comparação ao de núcleos monolíticos.

1.6.3. Arquitetura de micronúcleo:

A arquitetura de micronúcleo (*microkernel*) também é uma forma de arquitetura modular ou em camadas. Na tentativa de reduzir os procedimentos mais fundamentais, somente um pequeno número de serviços tais como parte do gerenciamento de memória, a sincronização entre processos e a comunicação entre processos, terá acesso direto ao HW, como representado na figura 1.2.

Por sua vez, o serviço de comunicação entre processo, que está dentro do micronúcleo, é o responsável por habilitar os serviços de, p.ex., redes, sistemas de arquivos, gerenciamento de dispositivos, etc., que normalmente, podem ser implementados no núcleo do sistema operacional, não no micronúcleo, ou até como procedimentos (aplicativos) externos ao núcleo.

Alguns sistemas operacionais permitem que as aplicações acessem diretamente os serviços oferecidos pelo micronúcleo. Por também ser uma arquitetura modular, a arquitetura em micronúcleo possui, em geral, as mesmas vantagens e desvantagens da arquitetura em camadas.



1.2. Representação da organização de um SO com arquitetura de microkernel.

1.7. A Função do Sistema Operacional

O sistema operacional é um programa que controla e coordena o uso do *hardware* do computador entre os vários programas de aplicação para os vários usuários. Assim, podemos dizer que o sistema operacional é um conjunto de módulos de *software* que regem os recursos do sistema, resolvem seus conflitos, simplificam o uso da máquina e otimizam seu desempenho global.

Um sistema operacional deve ter o completo domínio sobre os recursos da máquina. O escalonamento de recursos, o controle de entrada e saída (E/S), a gerência da memória, a gerência do processador, o escalonamento de processos e a segurança são funções que o sistema operacional deve exercer.

No projeto de um sistema operacional deve-se ter em mente dois objetivos principais:

- Apresentar ao usuário do computador uma forma amena de utilizar a máquina. Criar uma máquina virtual, de fácil compreensão para o usuário, com características diferentes da máquina física;
- Realizar o melhor uso possível do *hardware* disponível, aumentando o desempenho do sistema e diminuindo o custo.

É possível dividir as áreas de aplicação de um sistema operacional em **sistemas de tempo real** e **sistemas de processamento de tarefas**, que estão definidos a seguir.

17.1. Sistemas de Tempo Real

Os sistemas de tempo real são aqueles que devem fornecer uma resposta a estímulos externos num período de tempo extremamente pequeno. Podem-se citar os seguintes exemplos:

- **controle de processos:** Os computadores são utilizados para controlar processos industriais, tais como o refino de petróleo bruto, controle de temperatura em alto fornos entre outros. Em comum, tais aplicações têm a necessidade de receber uma resposta rápida após a emissão de um sinal de controle.

- **consulta a base de dados:** O computador aqui, é utilizado para obter informações armazenadas em grandes bancos de dados. Geralmente, o usuário deste tipo de aplicação desconhece como se processam as operações do sistema, e espera um tempo de resposta pequeno para obter suas informações. Um exemplo de tal aplicação seria uma consulta a informações sobre censo.
- **processamento de transações:** Neste caso, o computador é utilizado para realizar acessos a bancos de dados que estão frequentemente sendo atualizados, talvez várias vezes em um segundo. Aplicações típicas são reservas de passagens e consultas bancárias.

1.7.2. Sistemas de Processamento de Tarefas

Os sistemas de processamento de tarefas são projetados para manipular um fluxo contínuo de programas que serão executados pelo computador. Tendo em vista a grande variedade de programas que podem ser processados, tal sistema precisa suportar um grande número de utilitários. Esses utilitários podem ser compiladores para diversas linguagens, montadores (*assemblers*), *linkers*, editores de texto entre outros. É necessário também dar suporte a um sistema de arquivos para gerenciar o armazenamento de informações.

Podemos classificar os sistemas de processamento de tarefas em dois grupos:

- **Batch:** A principal característica desse grupo é o fato de que o usuário perde o controle do programa a partir do momento em que ele o submete ao sistema.
- **Interativo:** A característica marcante desse grupo é permitir a monitoração e o controle do programa, através de um terminal, enquanto durar o processamento.

1.8. A Evolução dos Sistemas Operacionais

Antigamente existia somente o *hardware* do computador. O operador e programador da máquina eram uma só pessoa. Todo o controle do sistema era feito através de botões e *displays* no console. O operador/programador monitorava a execução de um programa interativamente. A ativação (*setup*) do computador era muito lenta e a depuração era extremamente trabalhosa¹. Assim, era necessário buscar soluções que tornassem mais fácil e mais eficiente à utilização do computador. Os sistemas operacionais foram então criados com estas finalidades.

Os sistemas operacionais, assim como os dispositivos eletrônicos, vêm sofrendo mudanças ao longo das últimas décadas. Podem ser identificadas várias gerações, que estão descritas a seguir.

Primeira Geração (Anos 50)

Em 1953 surgiu o primeiro sistema operacional. Desenvolvido pela GM Laboratories, ele foi desenvolvido para um computador IBM modelo 701. Os primeiros

¹ Procure saber mais a respeito do funcionamento do ENIAC, o primeiro computador digital eletrônico, e ainda, sobre o Altair 8800, um dos primeiros computadores pessoais

sistemas eram voltados para o processamento em *batches* (lotes). O sistema operacional era responsável pela entrada de um programa e saída de outro, isto é, o sequenciamento de *jobs*. Uma vez que houvesse um processo rodando, ele tinha completo controle sobre a máquina. Ao término (normal ou anormal) do processo, o controle retornava ao sistema operacional que preparava a máquina para receber o próximo programa. Ainda assim, o tempo de ativação (*setup*) do computador era enorme.

Segunda Geração (Anos 60)

Nesta época, várias empresas já fabricavam sistemas operacionais. O principal objetivo era terminar o maior número de processos por unidade de tempo, aumentando o desempenho de um sistema de computador.

Foram desenvolvidos também nesta época os primeiros sistemas com as seguintes características:

- **Multiprogramação:** A multiprogramação permite que vários processos sejam executados simultaneamente.
- **Multiprocessamento:** No multiprocessamento, vários programas são processados ao mesmo tempo em processadores diferentes.
- **Time-sharing:** São ditos de tempo compartilhado. Usuários interagem com a máquina de uma maneira conversacional através de terminais.
- **Tempo real:** Dentre eles se destaca o SABRE para reservas de passagens da *American Airlines*.

Apareceu também nesta geração, o conceito de dispositivo independente. Nos sistemas da primeira geração, o usuário que precisasse escrever dados em uma fita, deveria referenciar, no seu programa, especificamente qual fita ele desejava. Na segunda geração, o usuário deveria apenas referenciar a necessidade de uma fita. O sistema, então, ficaria responsável por reservar uma fita disponível para ele.

Em abril de 1964, a IBM lançou a série de computadores System/360 que eram compatíveis em termos de arquitetura e tinham o mesmo sistema operacional OS/360. Esse sistema era adequado tanto para aplicações científicas quanto para aplicações comerciais. Antes disso, cada vez que fosse necessário um sistema computacional mais potente, eram oferecidos sistemas totalmente diferentes, o que implicava em uma conversão de *hardware* e de *software* lenta e muito cara. Com a série 360 era possível passar anos e anos sem precisar de conversões. Foi um verdadeiro sucesso.

Terceira Geração (meados dos anos 60 a meados dos anos 70)

A terceira geração começou efetivamente com a introdução da família 360. Os computadores dessa geração foram desenvolvidos para serem **sistemas de propósito geral**. Eram sistemas que suportavam simultaneamente o processamento *batch*, *time-sharing*, tempo-real e multiprocessado. Eram sistemas grandes e caros. Este conceito vendeu muitos computadores, mas existia um *overhead* muito alto relativo ao tempo em que a máquina gastava executando rotinas do sistema operacional. Além disso, determinadas aplicações não necessitavam de todos os modos de processamento.

Uma exceção deste conceito foi o sistema operacional UNIX, que foi construído nesta época. No final dos anos 60, Ken Thompson e Dennis Ritchie, entre outros membros da equipe do Bell Laboratories desenvolveram e implementaram um ambiente interativo, o UNIX. Usando esse sistema, desenvolveram também a linguagem C. Uma grande parte do sistema operacional foi escrita em C, o que contribuiu para a popularidade de ambos.

Os sistemas começaram a ser escritos em linguagens de alto nível e surgiram as linguagens de controle que permitem controlar a execução de processos.

Outro marco importante foi o surgimento da engenharia de *software*. Essa viria a ditar regras para a construção de sistemas. Os sistemas operacionais que eram um aglomerado de programas escritos por pessoas com muito mais noção de *hardware* do que *software* passaram a ser escritos de uma maneira mais disciplinada.

Quarta Geração (Meio dos anos 70 ao final dos anos 80)

Nesta fase apareceram os sistemas operacionais para redes de computadores, onde o usuário ganha acesso a redes locais ou geograficamente dispersas. O ponto importante era transferir informação entre computadores interconectados. Correio eletrônico, transferência de arquivo e aplicações de acesso a banco de dados proliferaram nesta época. O modelo **cliente/servidor** tornou-se difundido. Os **clientes** são os processos dos usuários que necessitam de vários serviços e os **servidores** são os componentes de *hardware/software* da rede que realizam estes serviços. Os servidores são geralmente dedicados a um tipo de tarefa tais como impressão, acesso a banco de dados entre outros.

O conceito de processamento distribuído tornou-se largamente difundido. Quando necessários, dados eram trazidos para serem processados em alguma instalação de computador central de larga escala.

Com o advento do microprocessador, surgem os computadores pessoais, um dos mais importantes desenvolvimentos com consequências sociais das últimas décadas. Um computador poderia ser adquirido por um preço acessível a muitos usuários que passariam a ter o seu próprio computador.

A quantidade de pessoas com acesso a um computador tornou-se consideravelmente maior e o termo *user friendly* começou a ser muito utilizado. Ele significa que o sistema apresenta, para usuários leigos, um ambiente de acesso fácil ao computador.

O conceito de máquina virtual tornou-se largamente utilizado. Os usuários não precisavam mais se preocupar com detalhes físicos do sistema do computador que está sendo usado, ao invés disso, o usuário via uma máquina virtual criada pelo sistema operacional. O campo da engenharia de *software* continuou a ter uma importância significativa.

Quinta Geração (Final dos anos 80 ao presente)

Nos anos 90 entramos na verdadeira era da **computação distribuída**. As computações são divididas em subcomputações. Essas são executadas em diferentes processadores, que podem ser computadores multiprocessadores ou redes de computadores. As subcomputações podem ser distribuídas de tal maneira que se possa obter vantagens utilizando computadores de propósito especial através das redes.

As redes podem ser configuradas dinamicamente. Elas continuam operando ainda que novos dispositivos e *softwares* sejam adicionados ou removidos. Quando cada novo servidor for adicionado, ele dará a rede, através de um procedimento de registro, informações sobre suas capacidades, políticas de ligação, acesso entre outras. Os clientes podem então usar os servidores, quando necessário, de acordo com os termos descritos durante o registro. Para atingir flexibilidade real, clientes não teriam conhecimento dos detalhes da rede.

Este tipo de conectividade é facilitada por padrões de sistemas abertos e protocolos. Esses padrões estão sendo desenvolvidos por alguns grupos internacionais como *International Organization for Standardization (ISO)*, *Open Software Foundation*, *X/Open*, entre outros. Eles pretendem chegar a um acordo sobre um ambiente internacionalmente aceito para padrões de comunicação e de computação. A tendência é a computação tornar-se muito poderosa e portátil.

Nos anos recentes, foram introduzidos os computadores *laptop*, que possibilitam às pessoas transportarem seus computadores por toda parte. Com o desenvolvimento de protocolos de comunicação os computadores *laptop* podem ser ligados em redes de comunicação e transmitir dados com alta confiabilidade.

1.9. Resumo

Os sistemas operacionais foram desenvolvidos ao longo das últimas décadas com dois propósitos principais. Primeiro, fornecer um ambiente conveniente ao desenvolvimento e a execução de programas. Segundo, controlar as atividades computacionais (recursos do *hardware*) para garantir um bom desempenho do sistema do computador.

Existem várias definições para os sistemas operacionais. Dentre elas, uma define os sistemas operacionais como um programa que atua como uma interface entre o *hardware* da máquina e o usuário do sistema.

Inicialmente os computadores eram manipulados através de consoles. Não havia nenhum mecanismo que facilitasse a sua utilização. Em uma fase posterior, foram desenvolvidos os montadores, carregadores, compiladores que melhoraram a tarefa da programação do sistema. Porém, o tempo de *setup* era grande, o que levou então ao desenvolvimento de sistemas em *batch*.

Depois desta época é que começaram a surgir os sistemas mais elaborados como os sistemas multiprogramados, multiprocessados, de tempo compartilhado e de tempo real. Um dos grandes destaques foi o conceito desenvolvido nos sistemas da IBM, a

série 360. Com essa série era possível mudar de um sistema para outro sem precisar de mudanças radicais.

Um outro destaque foi o desenvolvimento do sistema operacional UNIX, escrito em sua maior parte com uma linguagem de alto nível, o C. Mais recentemente surgiram os sistemas operacionais para rede de computadores. O conceito de processamento distribuído tornou-se difundido. Computadores pessoais tornaram-se acessíveis e possuem *softwares* amigáveis.

Atualmente, as redes de computadores têm obtido uma importância real. A computação distribuída vem sendo submetida a computadores multiprocessadores e a redes. O modelo cliente/servidor vem sendo desenvolvido, buscando portabilidade e melhor desempenho. Um outro tipo de computador que surgiu é denominado *laptop*. Os *laptops* são pequenos e podem ser transportados com facilidade².

² Existe alguma diferença entre notebooks e laptops? Faça uma pesquisa a respeito.