

Arquitetura Paralela

Há várias formas de se referenciar arquiteturas paralelas ou de multiprocessadores. Toda arquitetura paralela tem como principal objetivo aumentar o poder de processamento, aumentando o número de elementos de processamento.

Três características devem ser consideradas em uma arquitetura paralela:

- A granulosidade entre os elementos de processamento;
- A topologia que se refere ao padrão das conexões existentes entre os elementos de processamento;
- O controle de distribuição que está relacionado à alocação, à sincronização e à interação das tarefas dos elementos de processamento.

Na literatura existem duas taxonomias: a taxonomia de Flynn, que é a mais conhecida, e a taxonomia de Duncan, que abrange a maioria das arquiteturas paralelas.

Taxonomia de Flynn

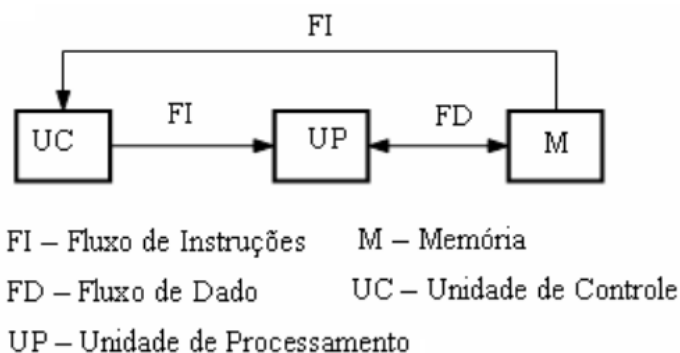
Foi proposta em 1966 e é bastante utilizada e respeitada. A classificação é baseada no fluxo dos dados propostos por Flynn. Esta classificação considera o processo computacional como a execução de uma sequência de instruções sobre um conjunto de dados. Aqui o modelo de Von Neumann, que corresponde à execução sequencial de instruções, é visto como um fluxo único de instruções, controlando um fluxo único de dados.

A introdução de fluxos múltiplos de dados ou de fluxos múltiplos de instruções é que faz surgir o paralelismo. A combinação de fluxos de instruções únicos e múltiplos com fluxos de dados únicos e múltiplos produz as quatro categorias de classificação de Flynn (1972).

		Fluxo de Dados	
Fluxo de Instruções	Único	SISD	SIMD
	Múltiplo	MISD	MIMD

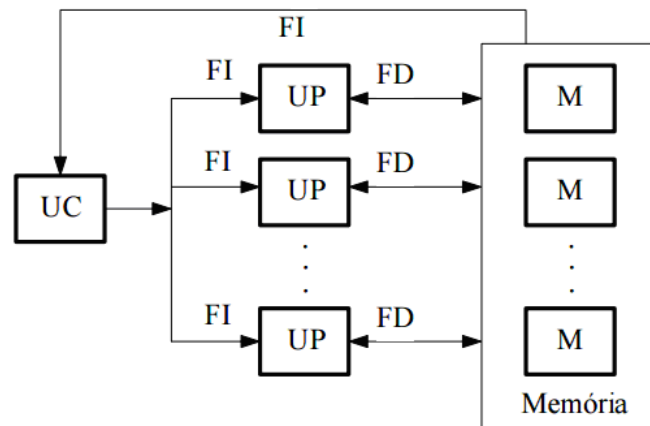
SISD – Single Instruction Stream/Single Data Stream

Fluxo Único de Instruções/Fluxo Único de Dados: Esta organização representa a maioria dos computadores sequenciais disponíveis atualmente (Princípio de Von Neumann). As instruções são executadas sequencialmente. Nessa categoria a execução sequencial, pode ser otimizada pelo uso de pipelines, no entanto utiliza uma única unidade de controle.



SIMD – Single Instruction Stream/Multiple Data Stream

Fluxo Único de Instruções/Fluxo Múltiplo de Dados: Há vários elementos processados (escravos) que estão sendo supervisionados por uma única unidade de controle (mestre). Todos os elementos processados recebem a mesma instrução para operar, mas recebem diferentes faixas de dados para processar. De acordo com abaixo. Normalmente os processadores vetoriais usam esse tipo de arquitetura paralela. Como é fácil de concluir, um computador com essa arquitetura é capaz de operar um vetor de dados por vez. Daí vem o nome de computadores vetoriais, ou Array Processor (Calônego, 1997).

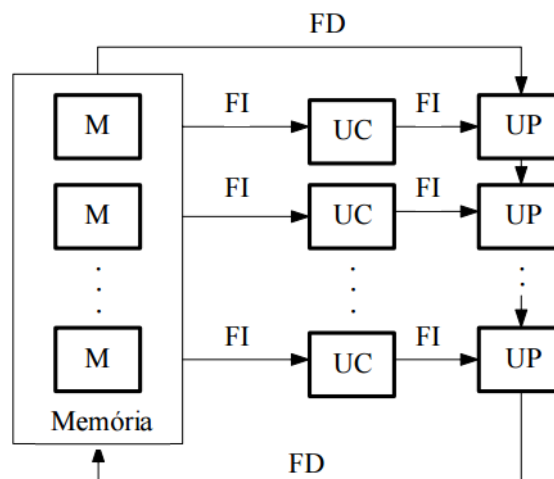


O programa de arquitetura paralela é armazenado em uma memória de controle, que é também o repositório inicial para os dados. Os dados que serão processados pelos elementos de processamentos devem ser distribuídos a diversos elementos de memória.

A forma de conexão entre cada elemento de processamento e cada elemento de memória caracteriza a arquitetura SIMD e o modo pelo quais os elementos de processamento trocam informação entre si. Se cada UP tem acesso exclusivamente a uma M, caracteriza a arquitetura SIMD de memória local, ou seja, uma rede de interconexão interprocessamentos. Por outro lado, se todos os UP estão conectados a todos os elementos M através de uma rede de interconexão, então a troca de informações é realizada por acesso a posições de memória compartilhadas, caracterizando assim um sistema SIMD de memória global.

Essa arquitetura é muito usada quando um mesmo programa deve ser executado sobre uma grande demanda de dados, como exemplo, a prospecção de petróleo. Nessa arquitetura não há problemas com a sincronização de tarefas, pois há um único programa em execução.

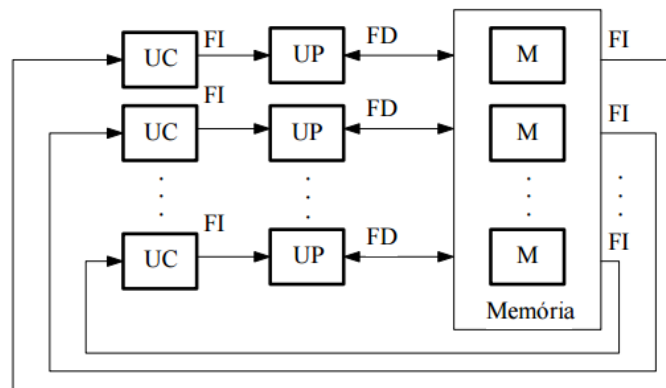
MISD – Multiple Instruction Stream/Single Data Stream



Fluxo Múltiplo de Instruções/Fluxo Único de Dados: Existe n UPs unidades de processamento, cada uma recebendo diferentes instruções sobre um mesmo conjunto de dados. Alguns autores classificam essa arquitetura pouco significativa e outros não a consideram. Em alguns casos, os “macropipelines” são colocados na categoria, uma vez que a saída de uma UP serve de entrada para outra UP.

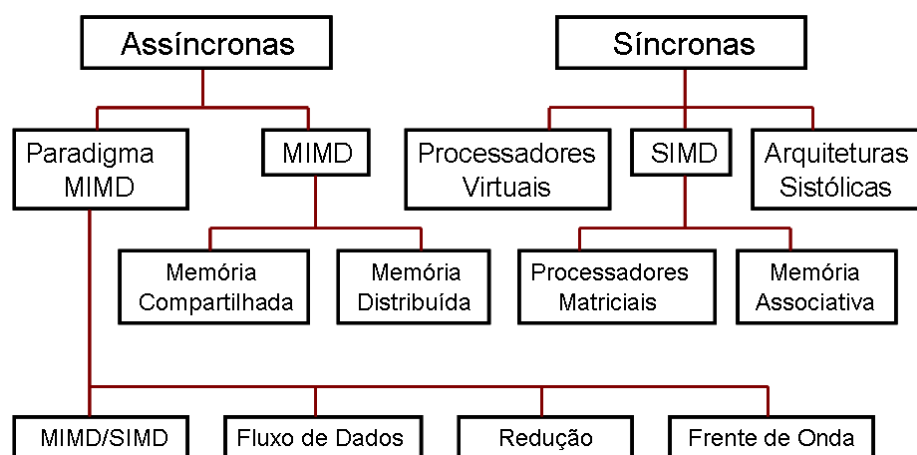
MIMD – Multiple Instruction Stream/Multiple Data Stream

Fluxo Múltiplo de Instruções/Fluxo Múltiplo de Dados: A maioria dos sistemas multiprocessados está incluída nesta categoria. Nesta classe cada elemento de processamento é controlado pela sua própria unidade de controle, executando instruções independentemente sobre diferentes fluxos de dados. Essa arquitetura apresenta uma grande flexibilidade para desenvolvimento de algoritmos paralelos.



Taxonomia de Duncan

Uma classificação mais refinada com o propósito de acomodar as inovações arquiteturais mais recentes, e que não se encaixa na de Flynn, foi proposta por Duncan. Essa nova taxonomia conserva a taxonomia de Flynn e divide as arquiteturas em síncronas e assíncronas, possibilitando a inclusão de arquiteturas mais recentes que não são absorvidas pela taxonomia de Flynn. A figura abaixo apresenta uma visão geral da classificação de Duncan (Alamas – Duncan, 1990).



Arquiteturas síncronas

Arquiteturas síncronas são aquelas que coordenam as operações concorrentes com base em sinais de relógio global em unidades de controle centralizado. Têm um clock para controlar os processos. O paralelismo vetorial é caracterizado pela execução da mesma operação sobre

todos os elementos de vetor de uma só vez, de forma a otimizar as operações que são efetuadas. Esses processadores fazem uso de pipeline (Duncan, 1990).

Assim como os processadores vetoriais, os processadores matriciais também se encaixam, na classificação de Duncan, dentro das arquiteturas síncronas. Por fazer parte das máquinas SIMD, operam com múltiplos processadores que trabalham de forma paralela e síncrona.

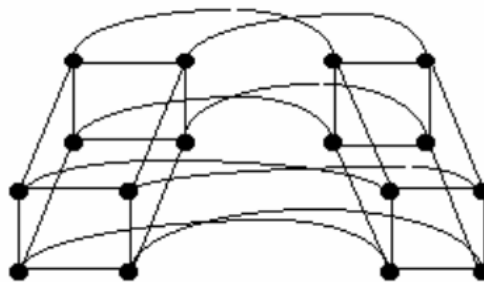
Outro tipo de arquitetura presente na categoria síncrona são os arranjos sistólicos. Nesse algoritmo para aplicações específicas, dados fluem da memória para a rede de processadores e voltam para a memória sincronamente. As informações fluem sincronamente como se fosse uma pulsação sanguínea.

Arquiteturas Assíncronas

As arquiteturas assíncronas, que constituem o outro grande grupo em que é dividida a classificação de Duncan, caracterizam-se pelo controle descentralizado do hardware e cada instrução é executada em elementos de processamento diferentes. Essa categoria é composta por máquinas MIMD, convencionais ou não.

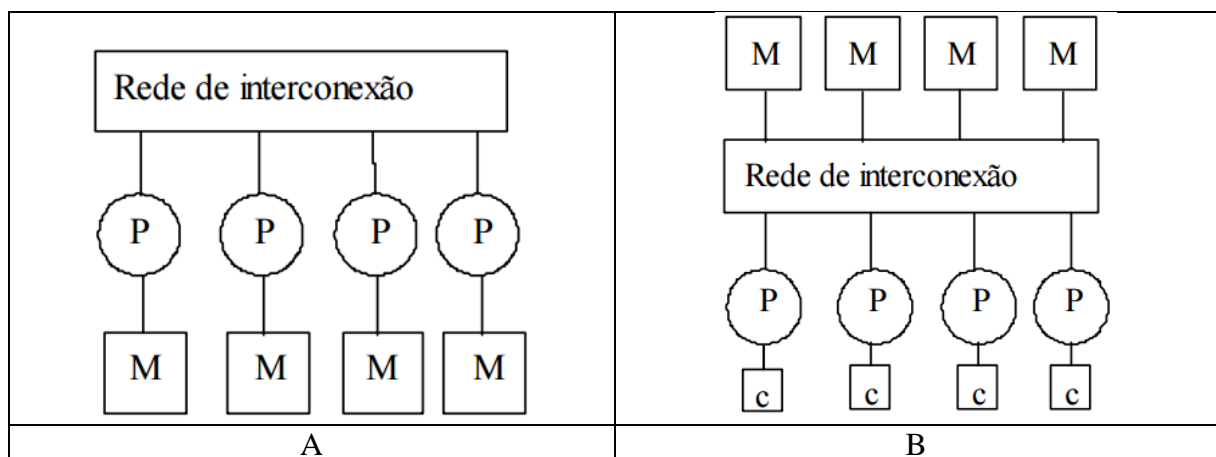
Arquitetura MIMD são computadores assíncronos, caracterizam-se pelo controle do hardware descentralizado. Nessa arquitetura a memória pode ser centralizada ou distribuída.

Diversas topologias de interconexão de rede podem ser utilizadas, destacando-se entre elas as arquiteturas de topologias em anel, árvore, hipercubo (figura abaixo) e reconfigurável.



Nas arquiteturas de memórias distribuídas, Figura A, cada processador possui sua memória local e os nós são interconectados através de redes de interconexão. Dessa forma, os nós se comunicam por meio de passagens explícitas de mensagens.

As arquiteturas de memória compartilhada, Figura B, por sua vez, provem uma única memória que é compartilhada pelos processadores.



Essas arquiteturas são também chamadas de multiprocessadores e não possuem problema no que diz respeito à troca de mensagens, forma pela qual são implementadas a comunicação e o sincronismo dos processos. Várias topologias de interconexão também são para esse tipo de arquitetura: crossbar, barramento e múltiplos estágios.

Como demais arquiteturas, enquadram-se as arquiteturas híbridas como fluxo de dados, redução e de frente de onda. Arquiteturas híbridas são aquelas em que partes de arquiteturas SIMD são controladas por arquiteturas MIMD.

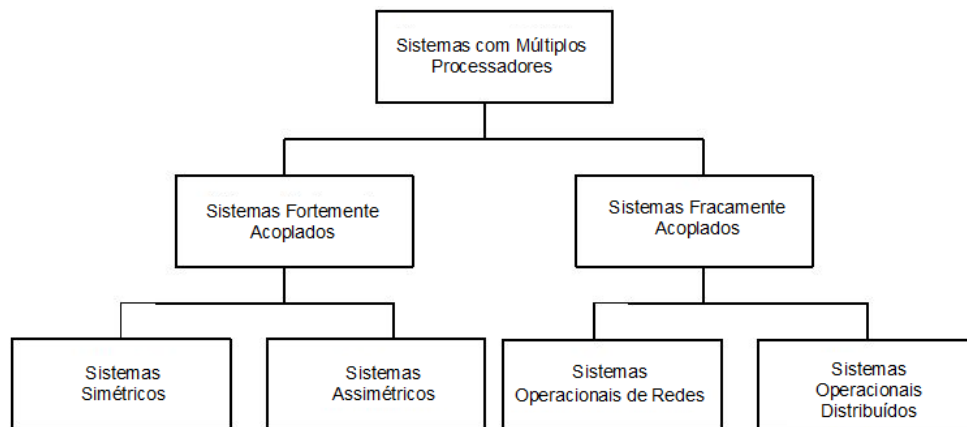
As arquiteturas Dataflow (ou fluxo de dados) compõem-se da execução de sequenciais de instruções baseadas na dependência de dados, que permitem a essas arquiteturas explorar concorrência em nível de tarefas, rotinas e até instruções.

Arquiteturas de redução, também conhecidas como dirigidas, apresentam um paradigma a partir do qual instruções são preparadas para executar quando os resultados são solicitados pelo operado ou por outras instruções. Redução implica a troca de parte do código-fonte pelo seu significado.

Finalmente, entre as arquiteturas que compõem a taxonomia de Duncan, têm-se as arquiteturas de frente de onda, que combinam pipeline de dados sistólicos com o paradigma de execução de Dataflow assíncrono.

Sistemas com Múltiplos Processadores

Os sistemas com múltiplos processadores caracterizam-se por possuir duas ou mais UCPS interligadas, trabalhando em conjunto. 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 em fortemente acoplados ou fracamente acoplados.

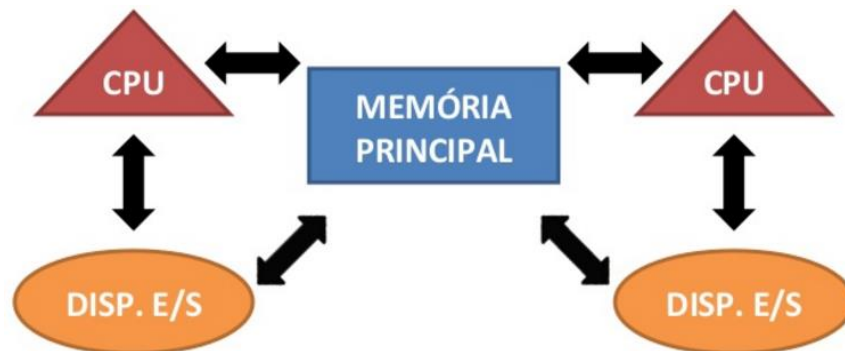


Sistemas Fortemente Acoplados

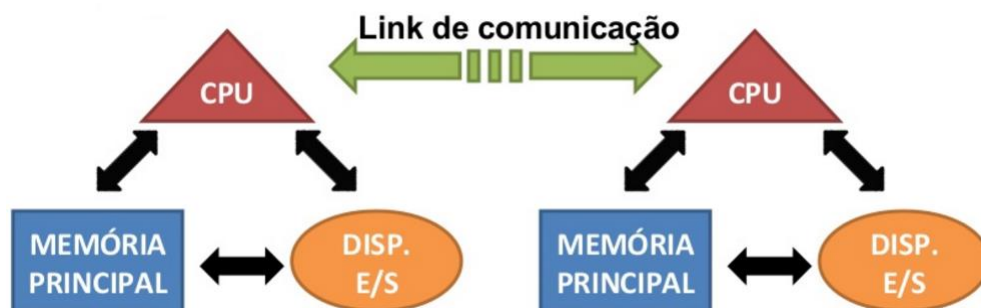
Em sistemas fortemente acoplados (*tightly coupled*) há vários processadores compartilhando uma única memória e gerenciados por apenas um SOp. Múltiplos processadores permitem que vários programas sejam executados ao mesmo tempo, ou que um programa seja dividido em subprogramas, para execução simultânea em mais de um processador. Dessa forma, é possível ampliar a capacidade de computação de um sistema, adicionando-se apenas novos processadores, com custo muito menor à aquisição de outros computadores.

Com o multiprocessamento, novos problemas de concorrência foram introduzidos, pois vários processadores podem estar acessando as mesmas áreas de memória. Além disso, existe o problema de organizar de forma eficiente os processadores, a memória e os periféricos.

Decorre do multiprocessamento o surgimento dos computadores, principalmente, para processamento científico aplicado, ex.: ao 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.



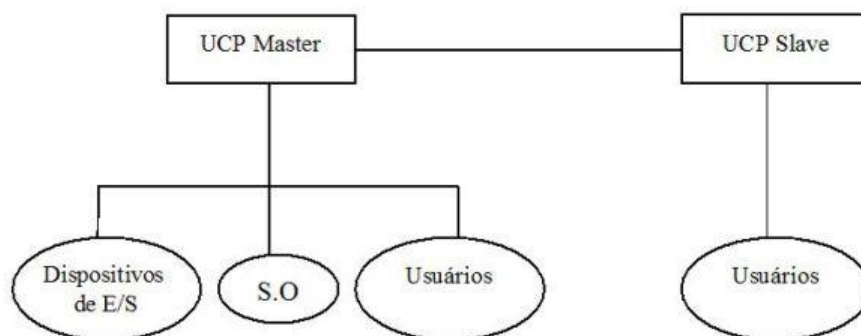
Sistemas Fortemente Acoplados



Sistemas Fortemente Acoplados

Sistemas Assimétricos

Na organização assimétrica ou mestre/escravo (master/slave), somente um processador (mestre) pode executar serviços do SOp, como, por exemplo, realizar operações de E/S. Sempre que um processador do tipo escravo precisar realizar uma operação de E/S terá de requisitar o serviço ao processador mestre. Dependendo do volume de operações de E/S destinadas aos processadores escravos, o sistema pode se tornar ineficiente, devido ao elevado número de interrupções que deverão ser tratadas pelo mestre.



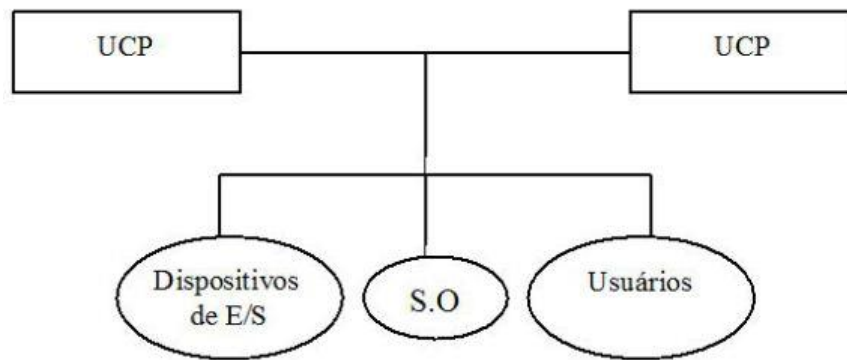
Sistemas Assimétricos

Se o processador falha, todo o sistema fica incapaz de continuar o processamento. Então, o sistema deve ser reconfigurado, fazendo um dos processadores escravos assumir o papel do mestre. Mesmo sendo uma organização simples de implementar e quase um extensão dos

sistemas multiprogramáveis, esse tipo de sistema não utiliza eficientemente o hardware, devido à assimetria dos processadores, que não realizam as mesmas funções.

Sistemas Simétricos

O multiprocessamento simétrico, SMP, ao contrário da organização mestre/escravo, implementa a simetria dos processadores, ou seja, todos os processadores realizam as mesmas funções. Apenas algumas poucas funções ficam a cargo de um único processador, como, por exemplo, a inicialização (boot) do sistema.



Sistemas Simétricos

Como vários processadores estão utilizando, independentemente, a mesma memória e o mesmo sistema operacional, é natural a ocorrência de acessos simultâneos às mesmas áreas de memória. A solução desses conflitos fica a cargo do hardware e do sistema operacional.

No processamento simétrico, um programa pode ser executado por qualquer processador, inclusive por vários processadores ao mesmo tempo (paralelismo). Além disso, quando um processador falha, o sistema continua em funcionamento sem nenhuma interferência manual, porém com menor capacidade de computação.

Os sistemas simétricos são mais poderosos que os assimétricos, permitindo um melhor balanceamento do processamento e das operações de entrada/saída, apesar de sua implementação ser bastante complexa.

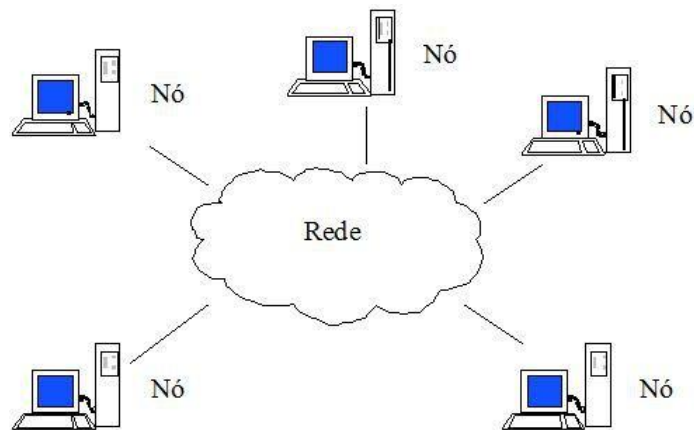
Sistemas Fracamente Acoplados

Os sistemas fracamente acoplados caracterizam-se por possuir dois ou mais sistemas de computação interligados, sendo que cada sistema possui o seu próprio sistema operacional, gerenciando os seus recursos, como processador, memória e dispositivos de entrada/saída.

Até meados da década de 1980, os sistemas operacionais e as aplicações suportadas por eles eram tipicamente concentradas em sistemas de grande porte, com um ou mais processadores. Nos sistemas centralizados, os usuários utilizam terminais não inteligentes conectados a linhas seriais dedicadas ou linhas telefônicas públicas para a comunicação interativa com esses sistemas.

No modelo centralizado, os terminais não têm capacidade de processamento. Sempre 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 de modelo de rede de computadores.



Sistemas Fracamente Acoplados

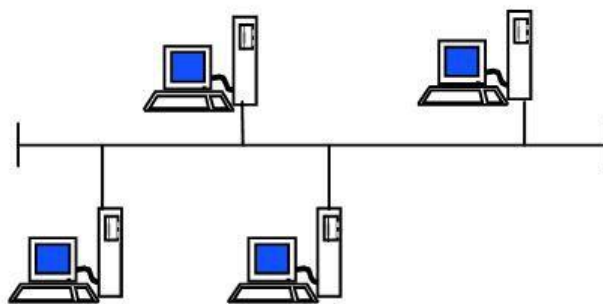
Sistemas Operacionais de Redes

Em sistemas operacionais de rede (SOR), cada nó possui seu próprio sistema operacional, além de um hardware e software que possibilitam ao sistema ter acesso a outros componentes da rede, compartilhando seus recursos. O SOR permite entre outras funções:

- Cópia remota de arquivos
- Emulação de terminal
- Impressão remota
- Gerência remota
- Correio eletrônico.

Cada nó é totalmente independente do outro, podendo inclusive possuir sistemas operacionais diferentes. Caso a conexão entre os nós sofra qualquer problema, os sistemas podem continuar operando normalmente, apesar de alguns recursos se tornarem indisponíveis.

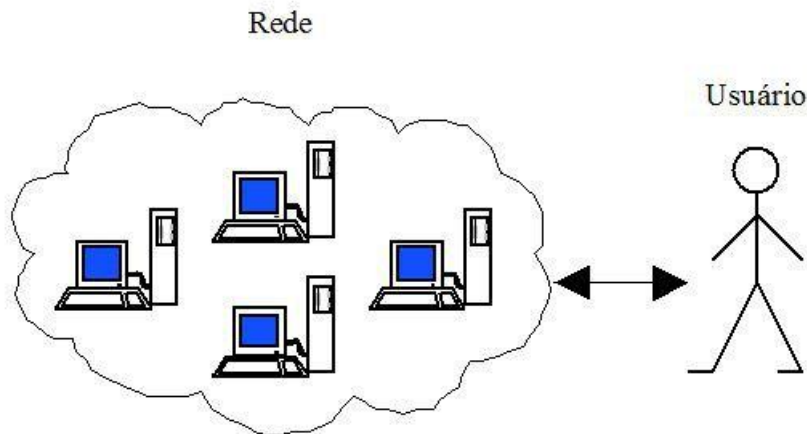
O melhor exemplo da utilização dos sistemas operacionais de rede são as redes locais. Nesse ambiente, cada estação pode compartilhar seus recursos com o restante da rede. Caso uma estação sofra qualquer, os demais componentes da rede podem continuar o processamento, apenas não dispondo dos recursos oferecidos por ela.



Sistemas Operacionais de Redes

Sistemas Operacionais Distribuídos

Em sistemas distribuídos, cada componente da rede também possui seu próprio sistema operacional, memória, processador e dispositivos. O que define um sistema distribuído é a existência de um relacionamento mais forte entre os seus componentes, onde geralmente os sistemas operacionais são os mesmos. Para o usuário e suas aplicações, é como se não existisse uma rede de computadores, mas sim um único sistema centralizado.



Sistemas Operacionais Distribuídos

A grande vantagem desses sistemas é a possibilidade do balanceamento de carga, ou seja, quando um programa é admitido para execução, a carga de processamento de cada sistema é avaliada e o processador mais livre é escolhido. Depois de aceito para processamento, o programa é executado no mesmo processador até o seu término. Também é possível o compartilhamento de impressoras, unidades de armazenamento, independentemente do sistema em que a aplicação esteja sendo processada. É muitas vezes chamado de cluster.

Suponha, por exemplo, uma configuração de dois computadores (COMP 1 e COMP 2), formando um cluster. Qualquer usuário conectado ao cluster poderá ter acesso aos dispositivos compartilhados, que permitem a ele imprimir uma listagem ou copiar um arquivo. Nesse tipo de configuração, se um dos sistemas falhar, o acesso aos dispositivos não será interrompido.

Os sistemas distribuídos podem ser considerados como uma evolução dos sistemas fortemente acoplados, onde uma aplicação pode ser executada por qualquer processador. Os sistemas distribuídos permitem que uma aplicação seja dividida em diferentes partes (aplicações distribuídas), que se comunicam através de linhas de comunicação, podendo cada parte ser processada em um sistema independente.

