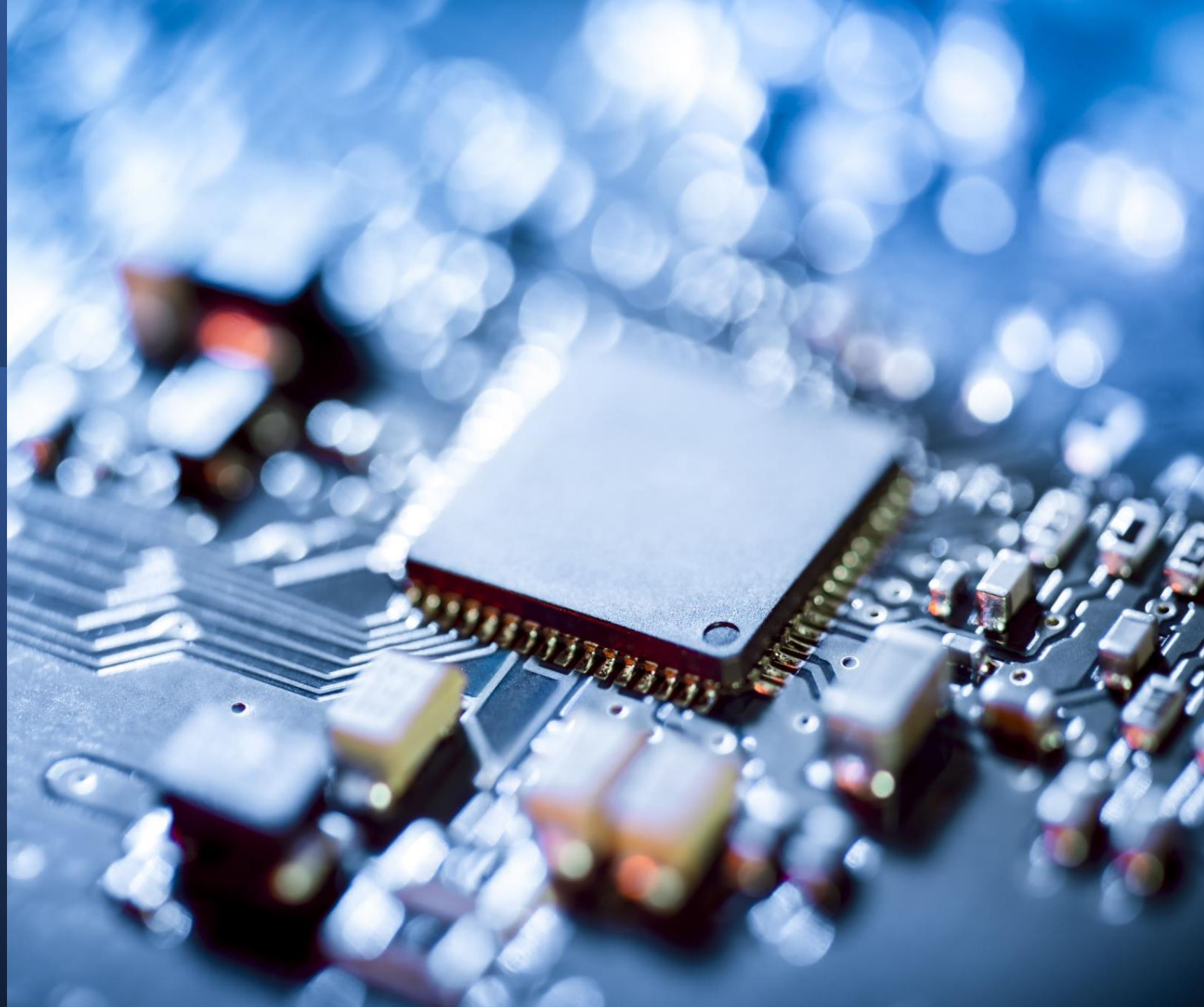


Professor Danilo

Sistemas com múltiplos processadores



Introdução

- **Sistemas com múltiplos processadores possuem duas ou mais UCP interligadas e que funcionam em conjunto na execução de tarefas independentes ou no processamento simultâneo de uma mesma tarefa.**
- Inicialmente os sistemas operacionais executavam um programa de cada vez mas com o surgimento dos sistemas operacionais com suporte a múltiplos processadores o processamento paralelo foi expandido.

Introdução

- **Os primeiros sistemas com múltiplos processadores surgiram nos anos 60 com o objetivo de diminuir o tempo de processamento para sistema que envolviam número elevado de cálculos complexos** ou que processavam alto volume de dados. Eram usados em aplicações científicas nas universidades ou em aplicações restritas do governo.
- **Foi nos anos 80 que os sistemas com múltiplos processadores começaram a ser usados amplamente pelas empresas.** Principalmente com a redução do custo do hardware ficou mais barato comprar sistemas com muitos processadores ou adicionar processadores a um sistema multi processado
- **Atualmente a maioria dos sistemas operacionais suportam multi processamento.**

A motivação para o uso de múltiplos processadores

Existem alguns fatores que motivaram o desenvolvimento de sistemas com múltiplos processadores:

- **O elevado custo para o desenvolvimento de processadores mais rápidos levou ao desenvolvimento de sistemas com múltiplos processadores** ao invés de sistemas com um único processador de alto desempenho.
- **Aplicações que requerem alto desempenho ou poder computacional.** As aplicações atuais cada vez mais demandam poder de processamento e desempenho que seriam dificilmente alcançados por uma arquitetura com um único processador com alto desempenho.

Os diversos benefícios dos sistemas com multi processadores

Inicialmente o desempenho foi o principal fator para o desenvolvimento dos sistemas multi processados porém posteriormente as empresas perceberam outros benefícios, antes não atingidos, que os sistemas multi processados oferecem.

- **Aumento da Confiabilidade** - Com mais de um processador, caso haja falha de hardware em um processador os demais processadores mantem os sistemas em funcionamento. Desta forma os sistemas multi processados aumentam a confiabilidade da infraestrutura de TI e dos serviços e sistemas da empresa.
- **Escalabilidade** - Se antes era necessário trocar o sistema computacional por um outro sistema computacional com mais poder de processamento, e isso custava muito caro, com os sistemas multi processados basta adicionar novos processadores conforme a demanda de processamento. Ou seja, o sistema multi processado é escalável, ele pode ser expandido conforme a necessidade de poder computacional da organização com a simples adição de novos processadores.

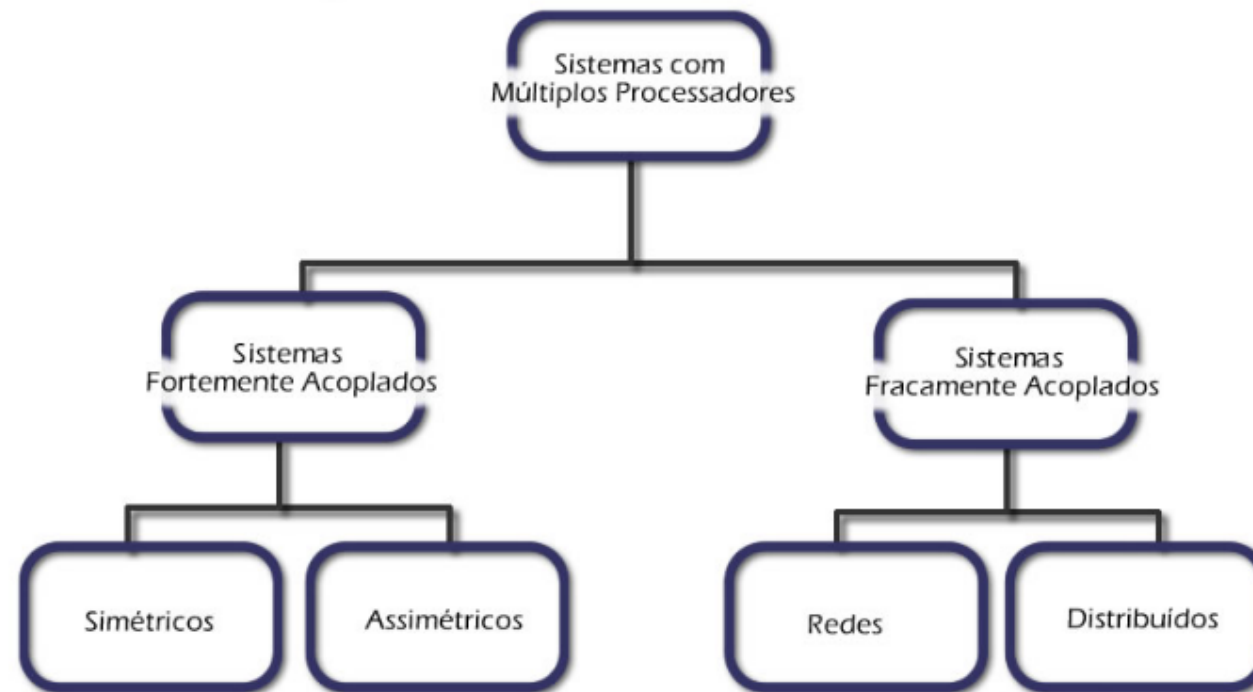
Os diversos benefícios dos sistemas com multi processadores

- **Alta disponibilidade** - Pelo mesmo motivo do aumento da confiabilidade, os sistemas multi processados oferecem alta disponibilidade dos serviços pois em caso de falha de um ou mais processadores, os outros processadores garantem a disponibilidade do serviço ou seja que o serviço não será interrompido.
- **Balanceamento de carga** - Todo o processamento pesado pode ser distribuído pelos vários processadores disponíveis no sistema, dividindo a carga do processamento e conseguindo melhores resultados em desempenho. Esta abordagem não seria possível em um sistema com um único processador de alto desempenho.

Atualmente os servidores de banco de dados, servidores de arquivos, servidores web usam sistemas com múltiplos processadores justamente para atender a estes requisitos de infraestrutura.

Tipos de Sistemas Computacionais

Tipos de Sistemas com Múltiplos Processadores.



Tipos de Sistemas Computacionais

Os sistemas são classificados conforme o grau de **paralelismo no processamento de instruções**. Um modelo proposto define quatro tipos de sistemas computacionais:

- **SISD (Single Instruction Single Data) - Sistemas que suportam uma única sequencia de instruções e apenas uma sequencia de dados.** Os sistemas com um único processador estão nessa categoria. Estes sistemas simulam paralelismo com a utilização de uma técnica conhecida como pipeline.
- **SIMD (Single Instruction Multiple Data) - Sistemas que com uma única sequencia de instruções tratam múltiplas sequencias de dados.** São sistemas que trabalham com vetores inteiros simultaneamente, possibilitando a execução uma mesma instrução ou bloco de instruções sobre diferentes elementos de um ou mais vetores.

Tipos de Sistemas Computacionais

- **MISD (Multiple Instruction Single Data)** - Sistemas que permitem a execução de múltiplas sequencias de instruções em uma única sequencia de dados. Não existe até o momento nenhum sistema computacional criado com esta arquitetura.
- **MIMD (Multiple Instruction Multiple Data)** - Sistema que permitem a execução de múltiplas sequencias de instruções sobre múltiplas sequências de dados. Esta categoria engloba os sistemas com múltiplos processadores.

Sistemas Fortemente acoplados e fracamente acoplados

- Os sistemas com múltiplos processadores usam a arquitetura MIMD e podem ser classificados em função de diversos fatores, como compartilhamento da memória principal, tempo de acesso à memória principal, mecanismos de comunicação e sincronização utilizados e velocidade de comunicação dos processadores.
- A partir destes fatores os sistemas com múltiplos processadores podem ser classificados em fortemente acoplados ou fracamente acoplados.

Sistemas Fortemente acoplados e fracamente acoplados

- ***Nos sistemas fortemente acoplados* os processadores compartilham a memória principal e são controlados por apenas um único sistema operacional.**
- ***Nos sistemas fracamente acoplados* temos dois ou mais sistemas computacionais independentes conectados por uma rede de comunicação, tendo cada sistemas seus processadores, memória principal, dispositivos E/S e sistema operacional.**
- A grande diferença entre estes dois sistemas é que em sistemas fortemente acoplados existe apenas um espaço de endereçamento compartilhado por todos os processadores, chamado de memória compartilhada. Nos sistemas fracamente acoplados cada sistema tem sua memória principal, seu espaço de endereçamento individual e a comunicação entre os sistemas é feita através de troca de mensagens.

Sistemas com Multiprocessadores Simétricos (SMP)

- **São sistemas fortemente acoplados que compartilham o mesmo espaço de endereçamento e são gerenciados por um único sistema operacional.** São também conhecidos como sistemas SMP.
- O tempo de acesso à memória principal pelos vários processadores é uniforme. Não importa a localização física do processador. Esta arquitetura é chamada por UMA (Uniform Memory Access).
- Sistemas SMP implementam a simetria dos processadores, onde todos os processadores realizam as mesmas funções. Poucas funções ficam a cargo de um processador central, como por exemplo, a inicialização do sistema. Como todos os processadores executam as mesmas funções existe um melhor balanceamento da carga e das operações de E/S.

Clusters

- **São sistemas fracamente acoplados formados por vários sistemas computacionais chamados nós que são conectados por uma rede que precisa ser de alto desempenho.** Cada nó da rede é denominado membro do cluster e possui seus próprios recursos como processadores, memória, dispositivos E/S e sistema operacional.
- Geralmente os membros do cluster são do mesmo fabricante por questões de incompatibilidade de sistemas operacionais.
- Como cada nó do cluster possui sua memória principal a comunicação entre os nós é feita através de troca de mensagens.
- Geralmente a rede de conexão é dedicada aos nós do cluster e o acesso aos serviços é feito através de uma outra rede.
- Este sistema oferece alta disponibilidade e tolerância a falhas.
- O usuário que acessa um cluster não tem conhecimento dos nós que compõem o cluster. Do ponto de vista do usuário é como se ele tivesse acessando um único sistema fortemente acoplado.

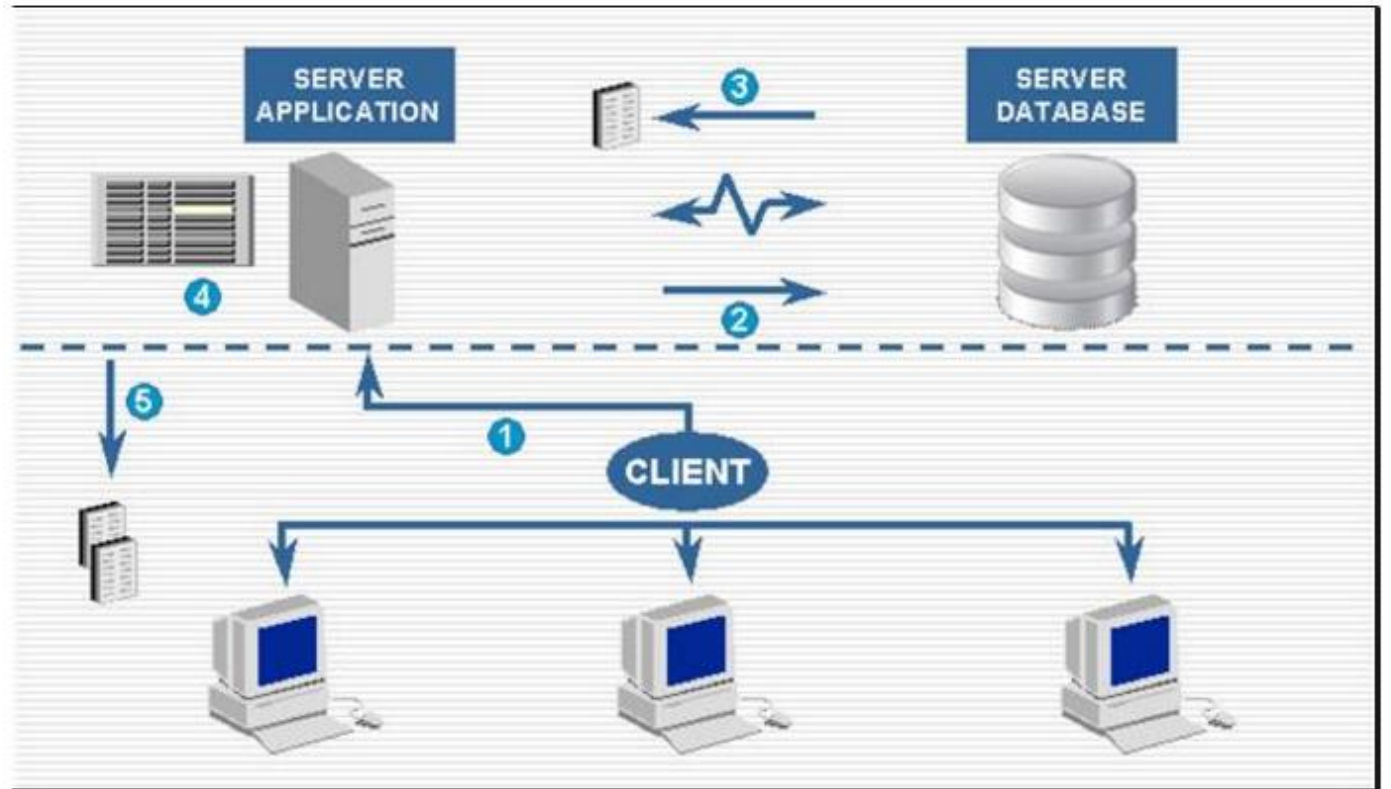
Sistemas Operacionais de Rede (SOR)

- **Os sistemas operacionais de rede são o melhor exemplo de um ambiente fracamente acoplado. Cada sistema, nó da rede, possui seus próprios recursos de hardware e são independentes dos demais nós da rede, sendo interconectados por uma rede formando uma rede de computadores.**
- Os SOR são usados tanto em redes locais como em redes Ethernet e redes distribuídas (WAN), sendo a comunicação feita através de uma interface de rede que possibilita o acesso aos demais componentes da rede.
- Não existe limite quanto ao número de nós que podem fazer parte da rede de computadores.
- Como cada nó possui seu próprio sistema operacional e os sistemas operacionais podem ser diferentes existe um protocolo que garante a comunicação entre os nós. No caso o protocolo TCP/IP.

Os SORs e o Modelo Cliente Servidor

- A grande maioria dos SORs e seus protocolos de rede implementa o modelo cliente-servidor.
- Neste modelo existe um ou mais servidores que oferecem serviços como servidor de impressão, servidor de arquivos, servidor de correio eletrônico, servidor de banco de dados. Os servidores oferecem os serviços aos clientes da rede.
- O Windows Server e o Novell Netware são exemplos de SORs voltados para oferecer este tipo de serviço.

Os SORs e o Modelo Cliente Servidor



Sistemas Distribuídos

- **Um sistema distribuído é um conjunto de sistemas autônomos que são interconectados por uma rede de comunicação e que funciona como se fosse um sistema fortemente acoplado mas de fato não é.**
- **Cada componente de um sistema distribuído possui seus próprios recursos e sistema operacional.**
- Os sistemas operacionais dos componentes que compõem um sistema distribuído podem ser heterogêneos.

Sistemas Distribuídos

- **O que diferencia um sistema distribuído dos demais sistemas fracamente acoplados é que existe um relacionamento mais forte entre seus componentes. Do ponto de vista físico, o hardware é independente, fracamente acoplado, porém do ponto de vista lógico, existe um relacionamento forte entre o software existente nos sistemas.**
- Os componentes de um sistema distribuído podem ser conectados por uma rede local ou através de uma rede distribuída. A localização de um componente não é considerada e nem as características físicas da rede.
- A escalabilidade de um sistema distribuído é, ao menos à princípio, ilimitada pois basta acrescentar novos componentes à rede em função da necessidade.

Sistemas Distribuídos

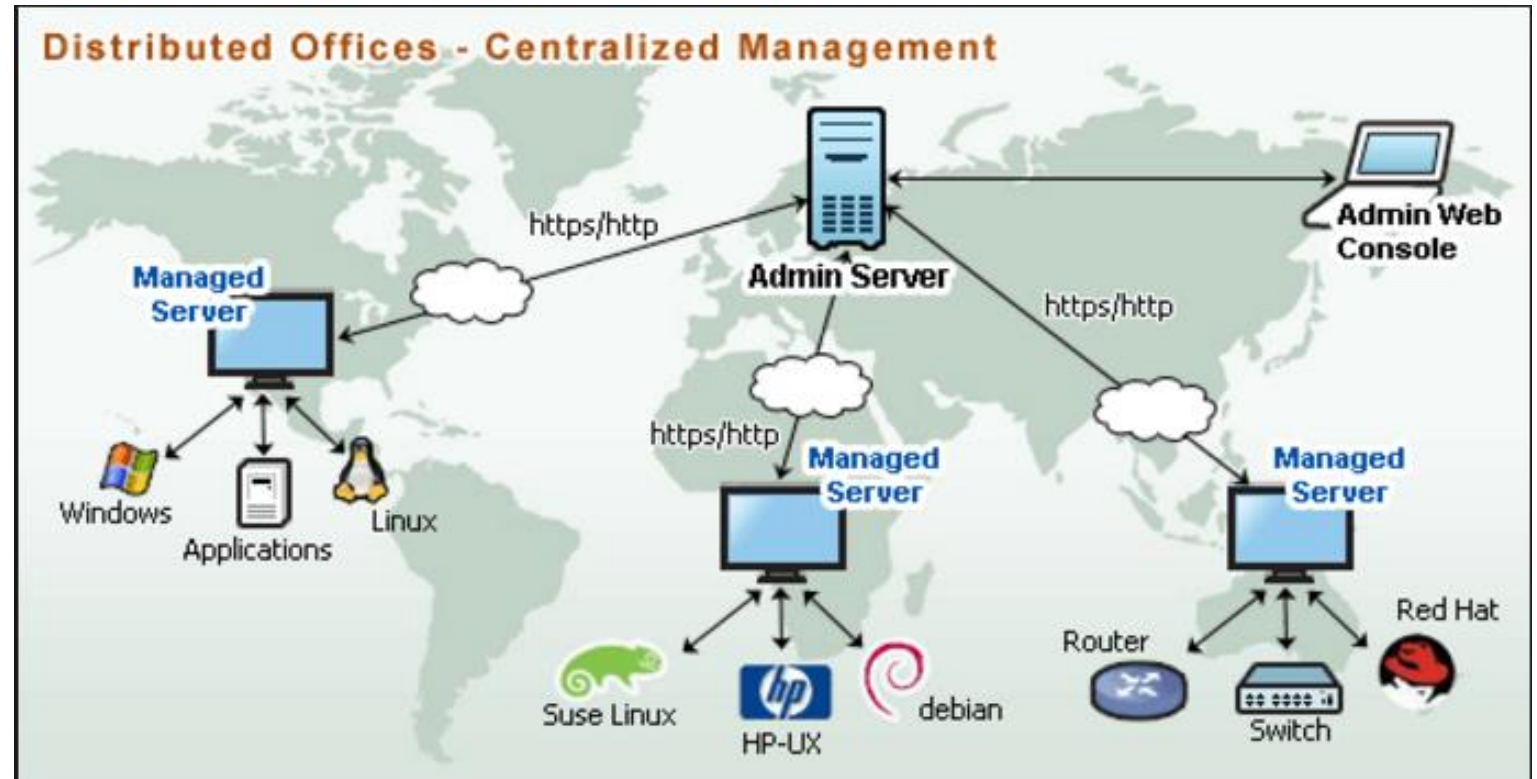


Imagem única do sistema

- Para o Usuário e suas aplicações é como se houvesse um único sistema fortemente acoplado ao invés de uma rede conectando sistemas heterogêneos e independentes.
- Este conceito é chamado de **imagem única do sistema**.

Transparência

- Em um sistema distribuído quando um usuário se conecta ao sistema não é necessário informar o nome do componente a que estará conectado. Independente da localização física dos componentes, o usuário terá acesso a todos os arquivos, diretórios e demais recursos de forma transparente.
- Ao executar uma aplicação o usuário nem mesmo saberá em quais componentes a sua aplicação está rodando. Caso ocorra um erro em um desses componentes o usuário não terá conhecimento ficando sob responsabilidade do sistema operacional a resolução de todos os problemas.
- **Este conceito é chamado transparência.** A partir deste conceito, o conjunto de componentes parece ser um sistema único, criando a imagem única do sistema.

Transparência

A transparência aqui é feita a partir de vários aspectos de um sistema distribuído:

- **Transparência de acesso**
- **Transparência de localização**
- **Transparência de concorrência**
- **Transparência de desempenho**
- **Transparência de escalabilidade**
- **Transparência a falhas**
- **Transparência de migração**

A tolerância a falhas

- Para que um sistema distribuído ofereça transparência em todos os aspectos acima mencionados é necessário que haja tolerância a falhas. Falhas tanto de hardware quanto de software. Neste caso o sistema tem que garantir que em caso de problema em algum componente as aplicações continuem sendo processadas sem qualquer interrupção do usuário, de forma totalmente transparente.
- A tolerância a falhas de hardware é garantida através de redundância de componentes como fontes duplicadas, vários processadores, técnicas de RAID, memória com detecção e correção de erros, etc. Ainda temos redundância dos meios de conexão, placas de rede, linhas de comunicação e dispositivos de rede.

A tolerância a falhas

- **A tolerância a falhas de software é mais difícil de implementar.** Caso haja uma falha no sistema operacional, a aplicação deve continuar como se não houve havido falha, sem que o usuário perceba que houve a falha. Neste caso, como as aplicações estão distribuídas em vários sistemas, caso ocorra algum problema com um dos componentes é possível que um deles assuma de forma transparente o papel do sistema defeituoso.
- **Por exemplo: Ao acessar um sistema de suporte houve falha na conexão com o banco de dados. O servidor de banco de dados, o serviço de banco de dados estava fora do ar. Neste caso o sistema de suporte se conecta a outro servidor de banco de dados, servidor de contingência e o sistema de suporte prossegue como se não tivesse havido falha na conexão ao banco de dados.**