

Conceitos e Projetos de Sistemas Distribuídos

Objetivos: Apresentar uma visão geral de processamento distribuído, analisando os tópicos mais importantes sobre o assunto. Entre esses tópicos estão: comunicação, sincronização, processos e tolerância a falhas. Discutir os problemas e abordagens nos projetos e desenvolvimento de Sistemas Distribuídos.

Introdução aos Sistemas Distribuídos

- De 1945 até 1985 temos computadores caros e grandes (microcomputadores custando dezenas de milhares de dólares)
- De 1985 em diante, o avanço da tecnologia mudou esta situação em dois pontos:
 - Desenvolvimento de poderosos microcomputadores (CPUs 8, 16, 32, 64 bits) custando uma fração do preço (1000 U\$ -- 10 milhões de instruções por segundo)

Introdução aos Sistemas Distribuídos

- Desenvolvimento de redes de computadores de alta velocidade
 - LANs (Local Area Networks) permitindo que até centenas de computadores dentro de um edifício sejam conectados, permitindo a troca de pequenas quantidades de informação.
 - WANs (Wide Area Networks) permitindo que milhões de computadores no mundo sejam conectados com velocidades de 64 Kbits/seg até Gigabits/seg.

O que é um Sistema Distribuído

Um *Sistema Distribuído* é uma coleção de computadores independentes que aparenta aos usuários do sistema como se fosse um único computador.

A definição tem dois aspectos:

- Hardware - as máquinas são autônomas
- Software - os usuários vêem o sistema como uma única máquina

Exemplos:

- Rede de estações de trabalho em uma Universidade
 - workstation do usuário
 - pool de processadores
 - único sistema de arquivos
 - sistema procura o melhor lugar para executar um determinado comando (própria estação, estação não sendo usada ou o pool de processadores)

Vantagens dos Sistemas Distribuídos Sobre os Sistemas Centralizados

A maior motivação para a descentralização é sem dúvida a questão econômica.

Lei de Grosch - O poder computacional de uma CPU é proporcional ao quadrado do seu preço. (Pagando 2 vezes mais você pode ter 4 vezes mais em desempenho).

A Lei de Grosch não é válida atualmente (é válida para tecnologia mainframe e não para microprocessadores).

Vantagens dos Sistemas Distribuídos sobre os Sistemas Centralizados

- Atualmente, a solução mais efetiva em termos de custo é o uso de um grande número de processadores baratos.
- Além disso, existe a possibilidade de conseguir um desempenho computacional que seria impossível usando processadores centralizados.

Vantagens dos Sistemas Distribuídos sobre os Sistemas Centralizados

- Outra razão para construir sistemas distribuídos é que algumas aplicações são inerentemente distribuídas. Exemplos:
 - sistemas de controle de produção de uma fábrica
 - sistema de reserva de passagens aéreas
 - sistema de suporte ao trabalho cooperativo

Vantagens dos Sistemas Distribuídos sobre os Sistemas Centralizados

- **Maior confiabilidade** - Distribuindo o trabalho para um conjunto de máquinas faz com que uma falha em uma máquina não pare o sistema todo (para sistemas críticos, usar sistemas distribuídos pode ser uma estratégia para se conseguir tolerância a falhas).

Vantagens dos Sistemas Distribuídos sobre os Sistemas Centralizados

- Escalabilidade (crescimento incremental)

Com sistemas distribuídos é possível acrescentar mais processadores ao sistema, permitindo expansão gradual na medida da necessidade.

Resumo das Vantagens dos SDs

Economia	Microprocessadores oferecem melhor fator preço/desempenho que os mainframes
Velocidade	SDs conseguem um maior poder computacional que os mainframes
Distribuição inerente	Algumas aplicações necessitam de máquinas separadas
Confiabilidade	Se uma máquina falha, o resto do sistema continua funcionando
Escalabilidade	Capacidade computacional pode ser adicionada , adicionando-se mais processadores

Vantagens dos SDs sobre PCs Independentes

Já que os PCs ficaram tão baratos porque não dar um para cada usuário trabalhar de forma independente? Não, devido:

- à necessidade de se compartilhar dados
- à necessidade de compartilhar periféricos caros
- à necessidade de termos comunicação pessoa a pessoa
- à flexibilidade de poder espalhar o trabalho existente em diversas máquinas, balanceando a carga de trabalho

Resumo das vantagens dos SDs. sobre PCs Independentes

Compartilhamento de dados	Permitir o acesso a bases de dados comuns
Compartilhamento de recursos	Permitir o acesso a periféricos específicos
Comunicação	Fazer a comunicação homem-homem mais fácil
Flexibilidade	Dividir a carga de trabalho

Desvantagens dos SDs

- Software
 - Falta de experiência no projeto, implementação e uso de software distribuído
 - Definição do quanto o usuário deve conhecer sobre a distribuição?
 - Definição do quanto deve o sistema fazer e quanto deve o usuário fazer?
 - Falta de um consenso entre os especialistas

Desvantagens dos SDs

- Rede de Comunicação
 - perda de mensagens
 - sobrecarga na rede
 - problema no dimensionamento da rede
- Segurança

Resumo das Desvantagens dos SDs

Software	Pouco software existente para SDs
Rede de Comunicação	Podem haver falhas ou saturação na rede de comunicação
Segurança	Facilidade de acesso pode criar problemas com dados sigilosos

Conceitos de Software

A imagem que o sistema apresenta ao usuário e o modo como eles pensam sobre o sistema é determinado principalmente pelo software.

S.O. para sistemas com múltiplas CPUs:

- Fracamente Acoplados
- Fortemente Acoplados

Software Fracamente Acoplado

Software Fracamente Acoplado permite que as máquinas e usuários do sistema sejam independentes uns dos outros, interagindo em um grau limitado quando for necessário.

Exemplo: Um grupo de computadores pessoais, cada um com a sua própria CPU, memória, disco rígido e S.O., compartilhando alguns recursos através de uma LAN.

Software Fortemente Acoplado

No Software Fortemente Acoplado os diversos processadores do sistema cooperam na execução das tarefas.

Exemplo: Processamento de imagens.

Sistema Operacional de Redes

(Software fracamente acoplado em hardware fracamente acoplado)

Exemplo: Rede de Estações de Trabalho conectadas por uma LAN

- cada usuário tem a sua estação de trabalho
- cada estação de trabalho pode ter ou não disco rígido
- cada estação de trabalho tem o seu próprio sistema operacional
- todos os comandos são normalmente executados localmente
- eventualmente é possível fazer uma conexão remota com outra estação de trabalho

Sistema Operacional de Redes (Cont.)

É provavelmente a combinação mais comum empregada na maioria das organizações.

Forma mais conveniente de comunicação e compartilhamento de informação: **Sistema de Arquivo Global Partilhado**

- Implementado em uma ou mais máquinas chamadas Servidores de Arquivos
- Servidores de Arquivos aceitam requisições de programa de usuários (chamados clientes)

Sistemas Distribuídos

(Software fortemente acoplado em hardware fracamente acoplado)

Objetivo do sistema é criar a ilusão para os usuários de que a rede de computadores é um único sistema timesharing

- Imagem de um sistema único
- Uniprocessador Virtual

Características dos SDs

- Um único mecanismo de comunicação interprocesso (global - qualquer processo pode se comunicar com qualquer outro)
- Esquema global de proteção
- Gerenciamento de processos precisa ser o mesmo no sistema todo (criação, destruição, iniciação, interrupção de processos)
- Único conjunto de chamadas do sistema (e que tenha sentido em um SD)

Características dos SDs (Cont.)

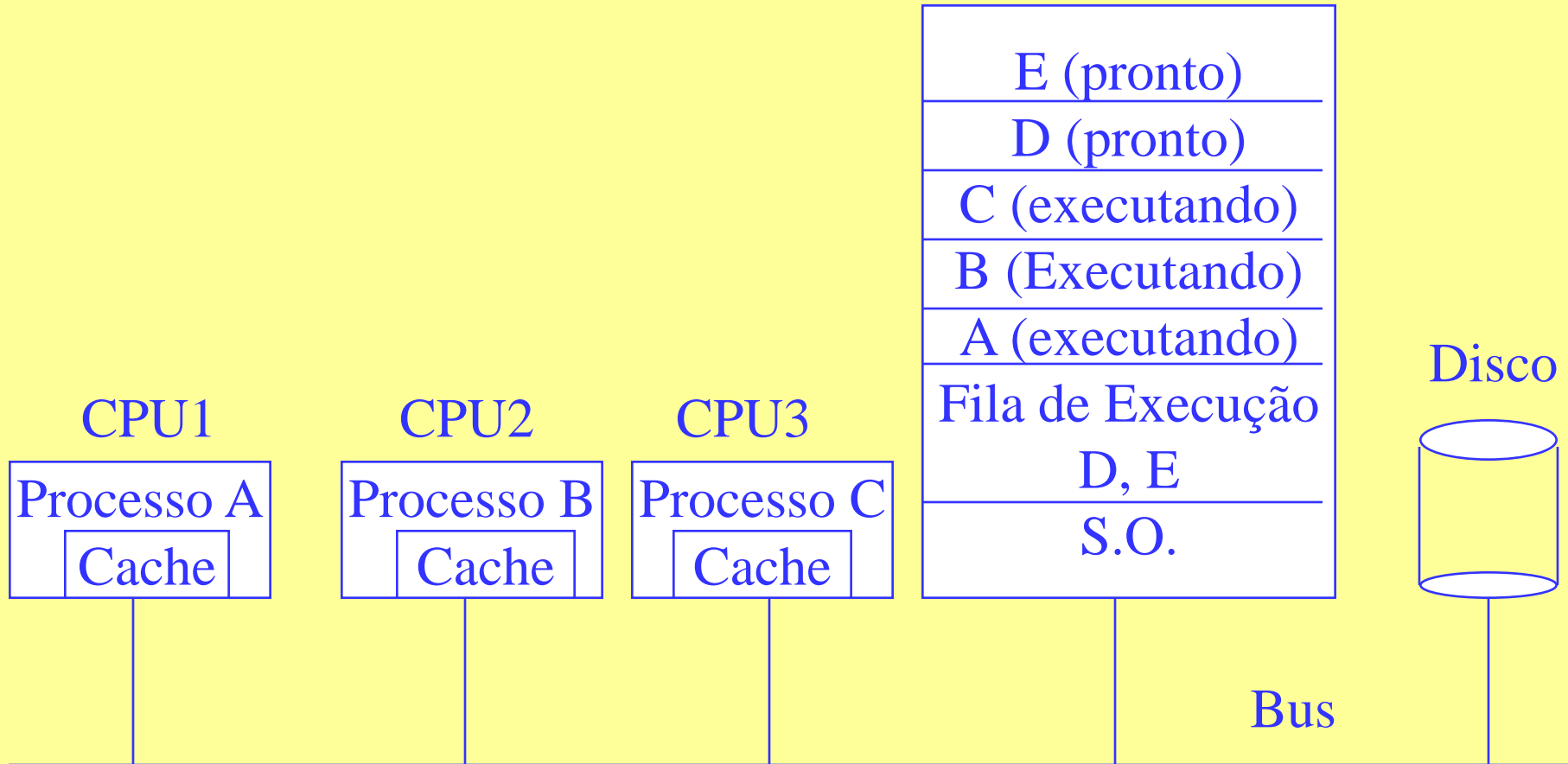
- Sistema de arquivo também precisa ter as mesmas características
- Cópias idênticas do kernel executam em todas as CPUs do sistema (escalonamento, swapping, paginação, etc)

Sistemas Timesharing Multiprocessadores

(Software Fortemente Acoplado em Hardware Fortemente Acoplado)

- Muitas máquinas dedicadas nesta categoria (bases de dados, processamento de imagens)
- Todo o projeto pode ser centralizado
- Existência de uma única fila de execução (processos que não estão bloqueados e prontos para execução)
- Memória compartilhada

Sistemas Timesharing Multiprocessadores



Diferença entre 3 tipos de Sistemas

	SO Rede	SO Distrib.	SO Multip.
Ele se parece com um uniprocessador virtual?	Não	Sim	Sim
Tem o mesmo SO?	Não	Sim	Sim
Número de cópias do SO	N	N	1
Como é a comunicação	Arquivos Partilhados	Mensagens	Memória Partilhada
Protocolos requeridos?	Sim	Sim	Não
Uma única fila de execução	Não	Não	Sim
Arquivos tem a mesma semântica?	Usualmente Não	Sim	Sim

Tópicos de Projetos

Transparência.- Pode ser conseguida em dois níveis diferentes:

1 Esconder a distribuição do usuário

Exemplo: Make no UNIX

2 Em um nível mais baixo (interface das chamadas do sistema pode ser projetada de tal forma que a existência de múltiplos processadores não seja visível)

Diferentes Tipos de Transparência

Locação	O usuário não pode dizer onde os recursos estão alocados
Migração	Recursos podem ser movidos sem troca de nomes
Replicação	Os usuários não sabem quantas cópias existem
Concorrência	Múltiplos usuários podem partilhar recursos automaticamente
Paralelismo	Atividades podem acontecer em paralelo sem o conhecimento dos usuários

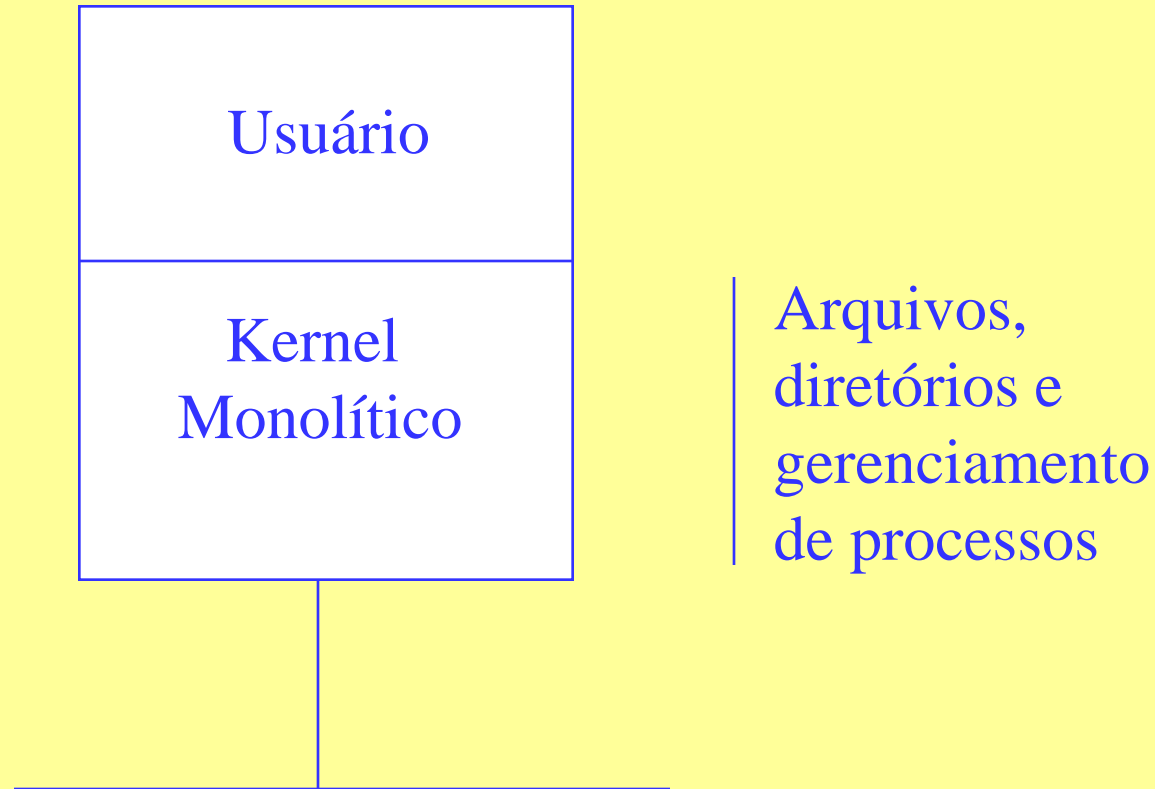
Flexibilidade

Como a tecnologia ainda não está consolidada, a flexibilidade é importante como um meio para caminhos alternativos.

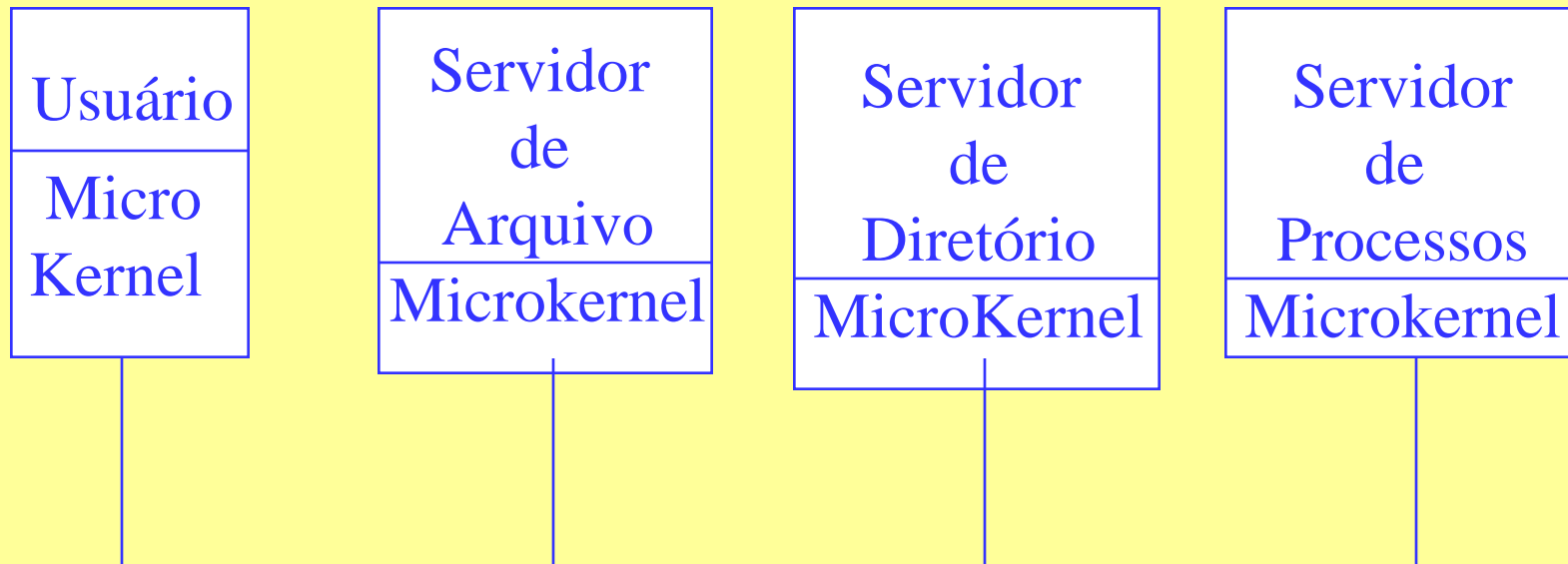
Existem dois modos de estruturar um SD:

- kernel monolítico
- Microkernel

Kernel Monolítico



Microkernel



Microkernel

Microkernel é mais flexível porque ele faz o mínimo possível:

- Comunicação entre processos
- Gerenciamento de Memória
- Gerenciamento de Processo (baixo nível) e Escalonamento
- E/S baixo nível

Confiabilidade

Teoria - se uma máquina falha, alguma outra realiza a sua tarefa

Prática - O funcionamento do sistema conta com a cooperação de alguns servidores específicos

Disponibilidade - fração de tempo que o sistema é “usável”. Disponibilidade pode ser aumentada por meio de redundância: pedaços de hardware ou software podem ser replicados

Quanto maior o número de cópias de software, maior a probabilidade de inconsistências.

Desempenho

O desempenho de uma particular aplicação executando em um SD não deve ser pior que em um sistema centralizado.

Medidas de desempenho:

- Tempo de resposta
- Throughput (número de jobs por hora)
- Capacidade consumida de rede

Desempenho

Comunicação é relativamente lenta (1 mseg para enviar uma mensagem e receber uma resposta)

Paralelismo de granularidade fina - grande número de pequenas computações, altamente interativas umas com as outras;

Paralelismo de granularidade grossa - grandes computações, poucas interações, poucos dados.

Escalabilidade

O sistema pode funcionar bem com centenas de processadores e falhar para milhares de processadores.

Características dos Algoritmos Descentralizados:

- Nenhuma máquina tem informação completa sobre o estado do sistema
- Decisões são tomadas baseadas em informação local
- Falha em uma máquina não compromete o algoritmo
- Não existe um relógio global

Gargalo dos Grandes Sistemas

	Exemplo
Componentes Centralizados	Um único servidor de mail para todos os usuários
Tabelas Centralizadas	Uma única lista telefonica on-line
Algoritmos Centralizados	Roteamento baseado em informação completa