Características de Sistemas Distribuídos

Carlos Ferraz

cagf@cin.ufpe.br

Tópicos

- O conceito de Sistemas Distribuídos
- Infra-estrutura básica
- Exemplos
- Vantagens e desvantagens
- Convergência digital
- Características

O Conceito de Sistema Distribuído

- "Coleção de computadores independentes que aparecem para os usuários do sistema como um único computador." (**Tanenbaum**)

- "Um sistema em que componentes de *hardware* e *software* localizados em computadores em rede se comunicam e coordenam suas ações por passagem de mensagens." (**Coulouris** *et al*)

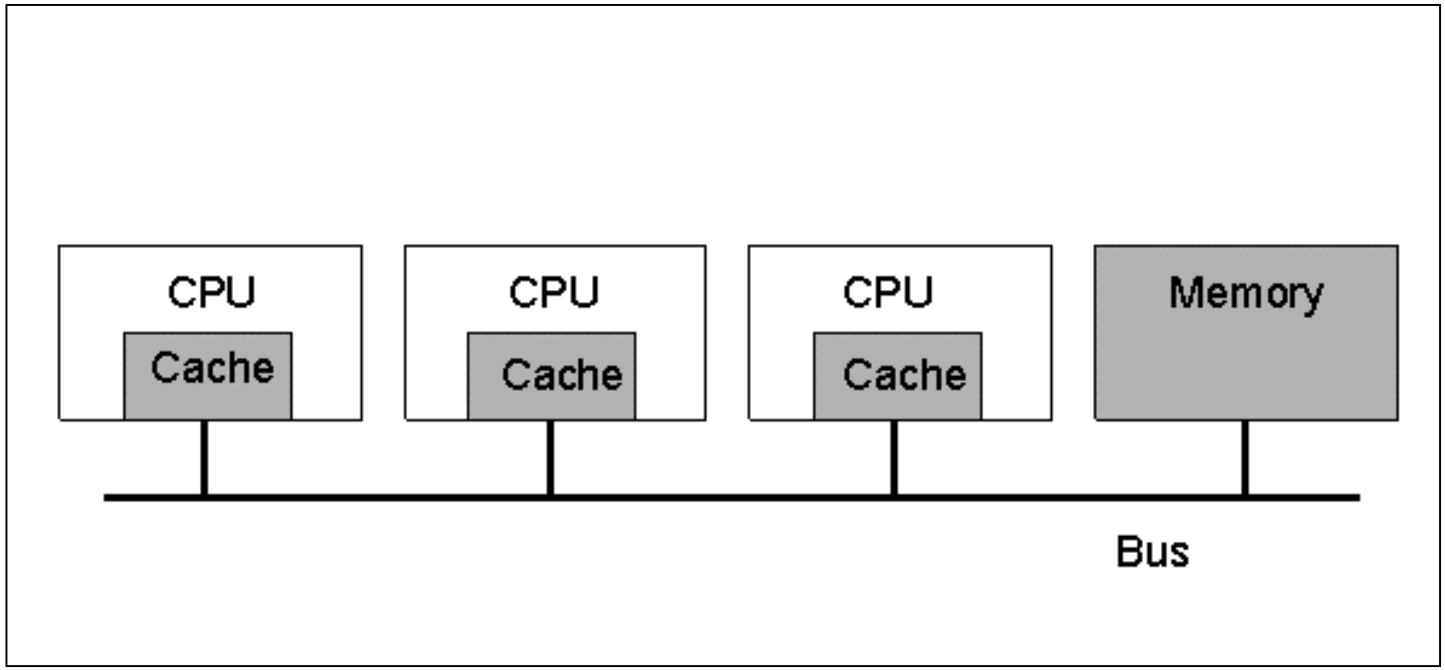
- Vários componentes
- Conectados via uma rede
- Compartilhando recursos

A infra-estrutura básica para Sistemas Distribuídos

- Rede
  - Com fio: velozes e confiáveis
  - Sem fio: cada vez mais velozes e ainda pouco confiáveis
- Hardware
- Software

Sistemas Distribuídos

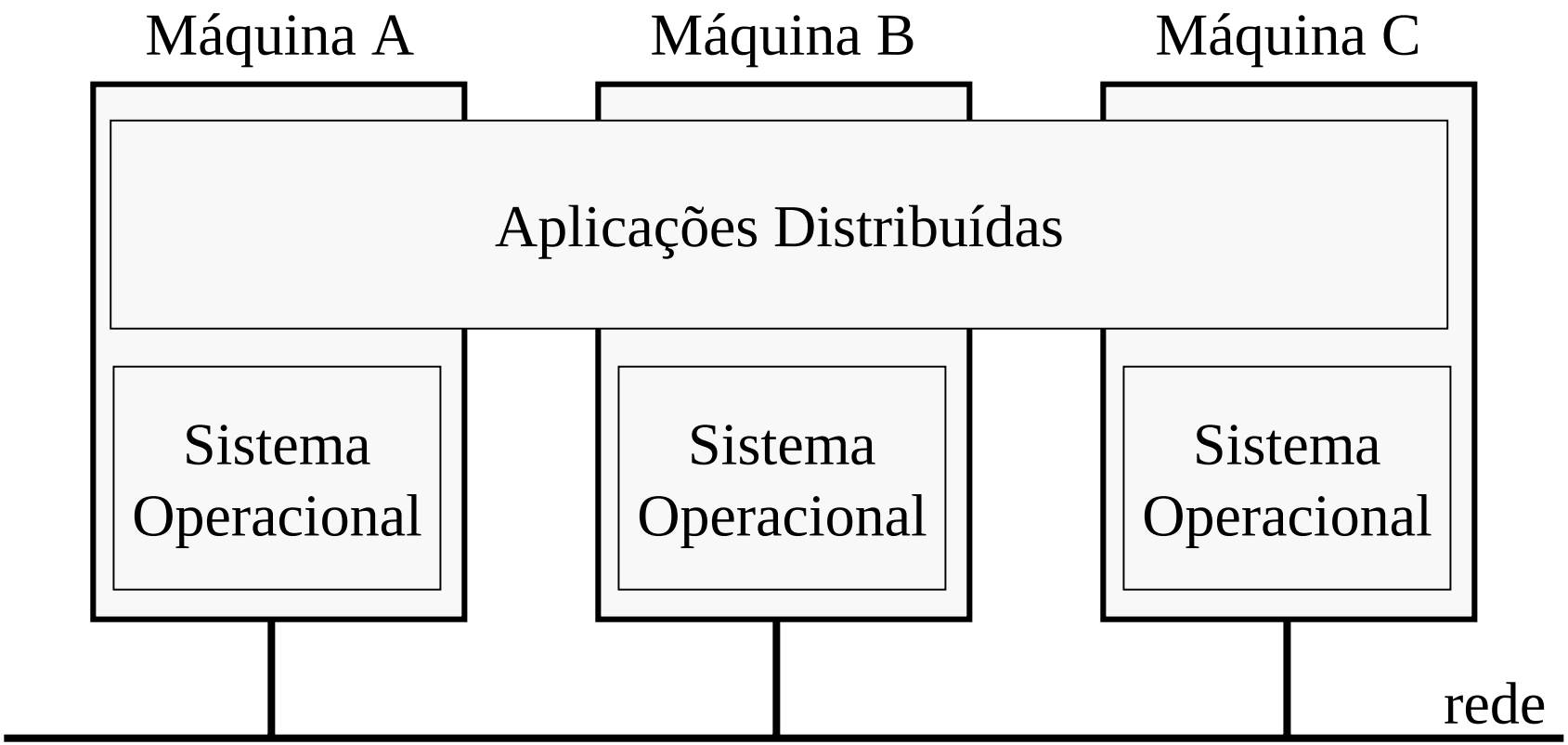
- Do ponto de vista de hardware
  - Multiprocessadores, com memória compartilhada



Tanenbaum and van Steen. Distributed Systems: Principles and Paradigms
© Prentice Hall 2002

Sistemas Distribuídos

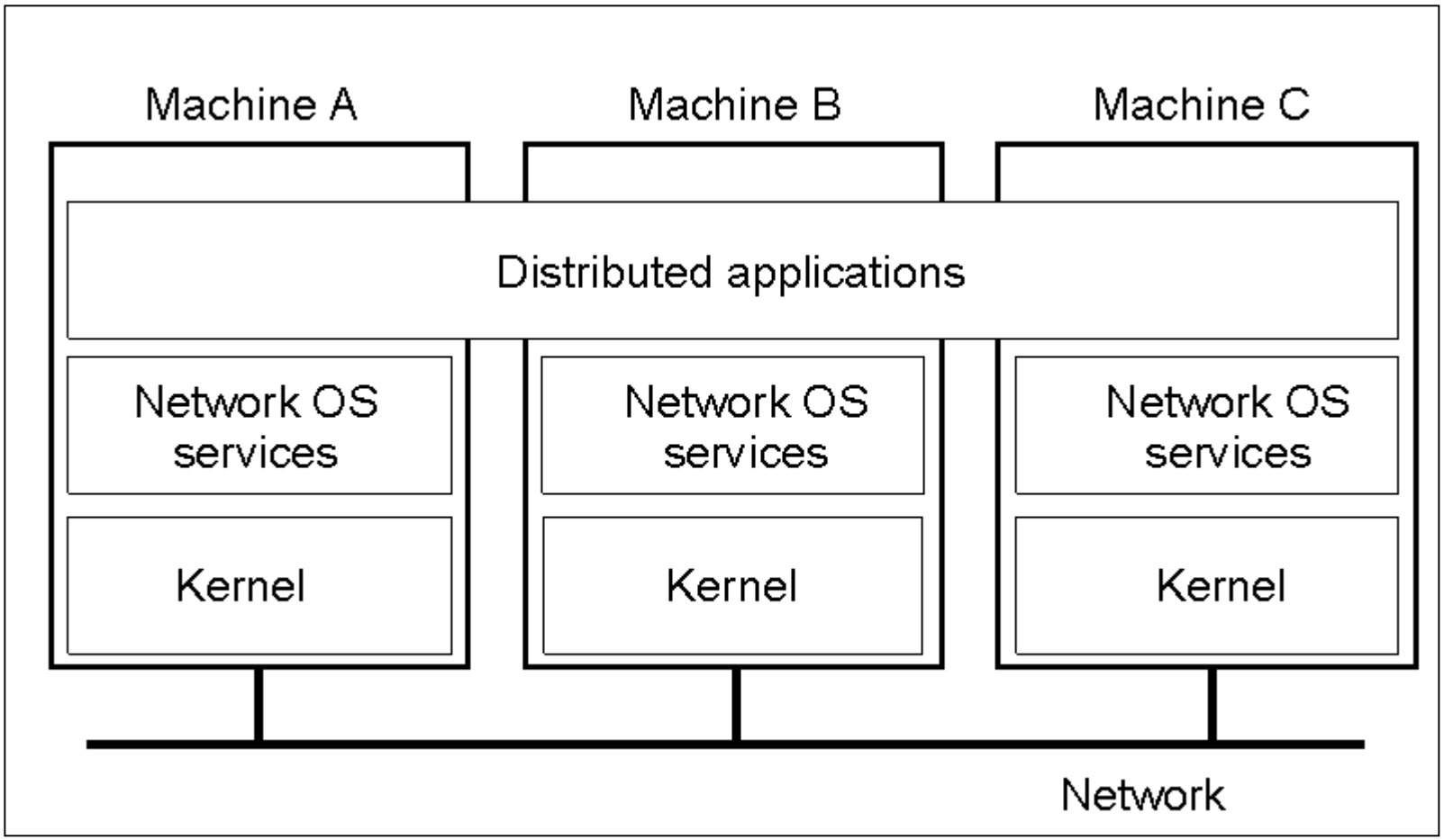
- Do ponto de vista de hardware
  - Multicomputadores, com memória privada



Sistemas Distribuídos

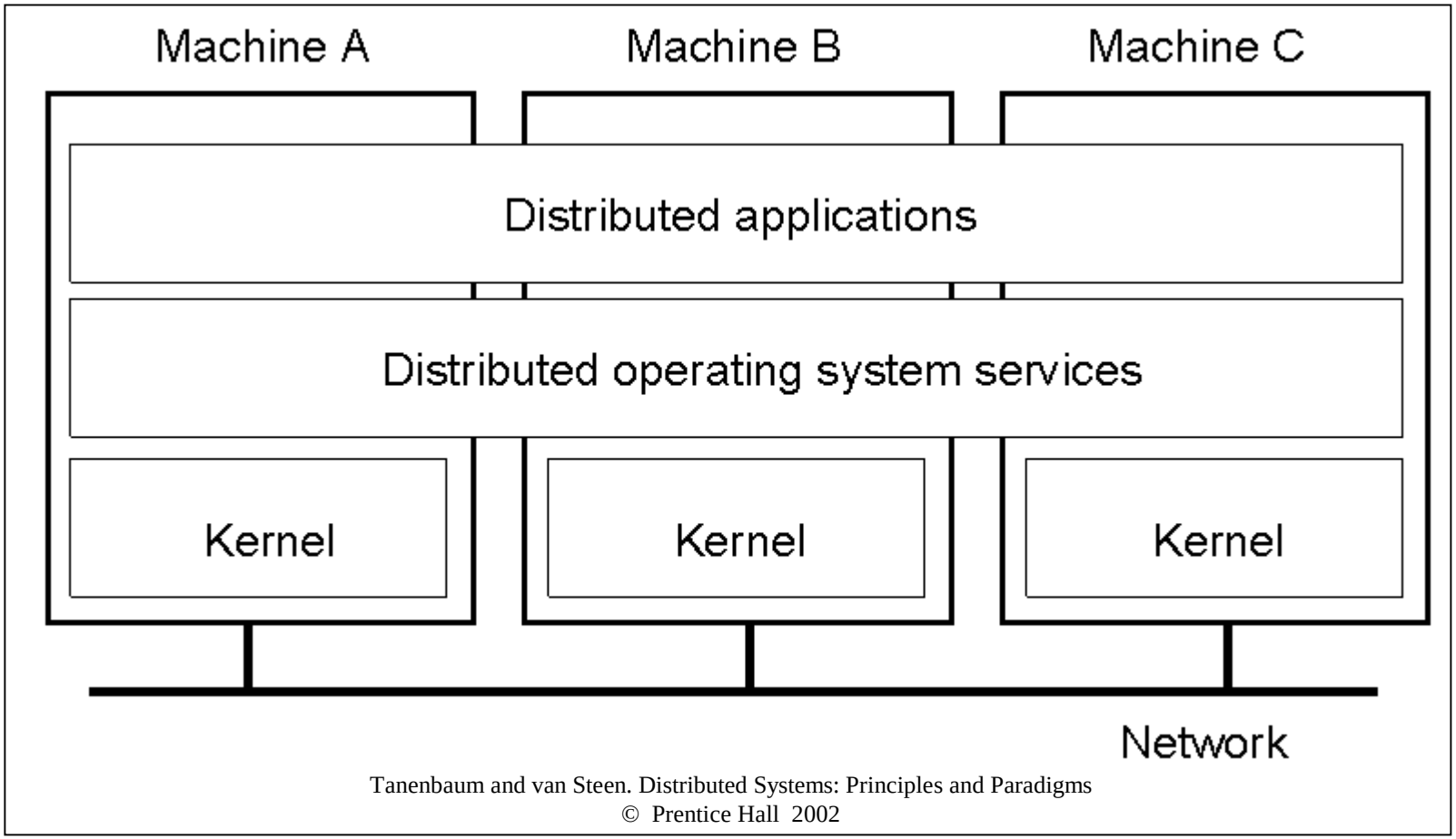
- Do ponto de vista de software
  - Sistemas operacionais de rede (SOR)
    - Exs.: Unix, Windows NT, 2000, XP
  - Sistemas operacionais distribuídos (SOD)
    - Exs.: Chorus, Mach, Amoeba
  - *Middleware*
    - Exs.: Java RMI, DCOM/MS, CORBA/OMG

Sistemas Operacionais de Rede



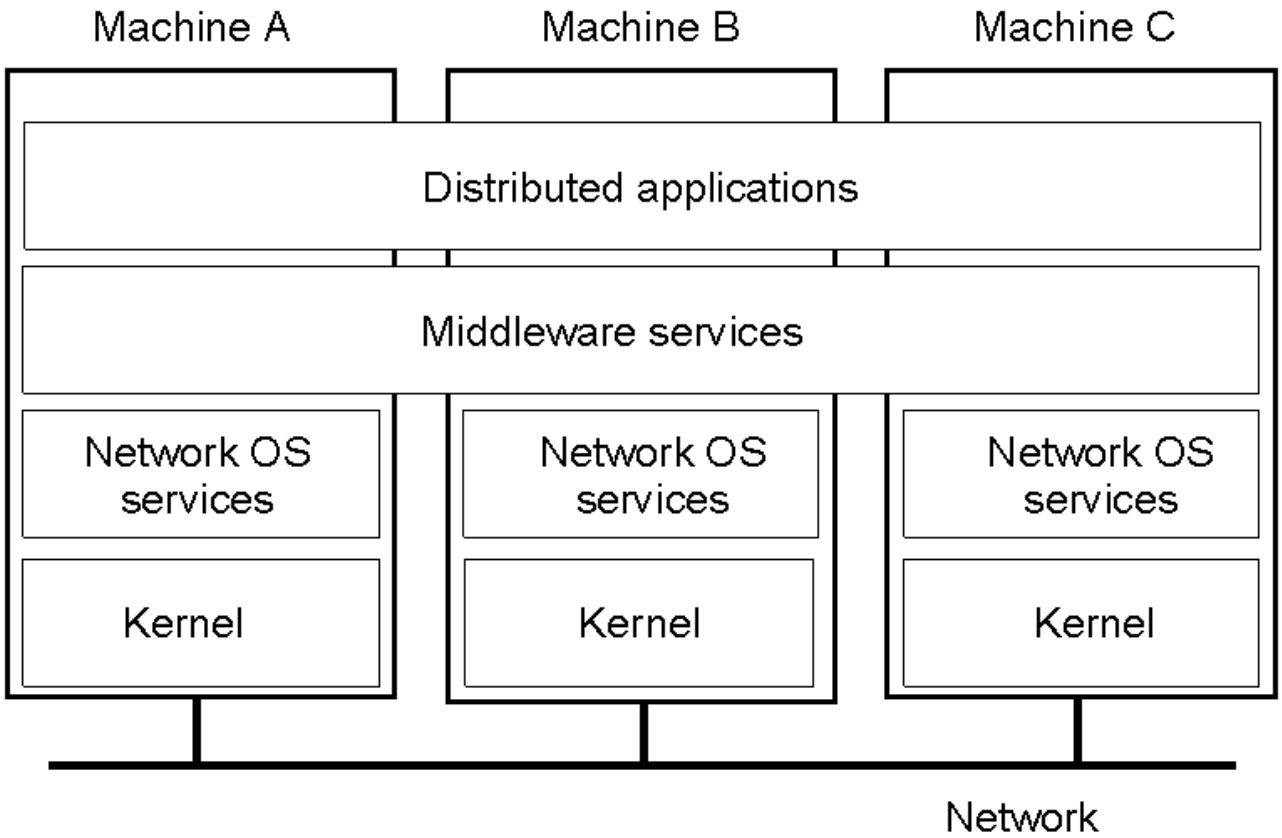
Tanenbaum and van Steen. Distributed Systems: Principles and Paradigms © Prentice Hall 2002

Sistemas Operacionais Distribuídos



Tanenbaum and van Steen. Distributed Systems: Principles and Paradigms
© Prentice Hall 2002

Middleware



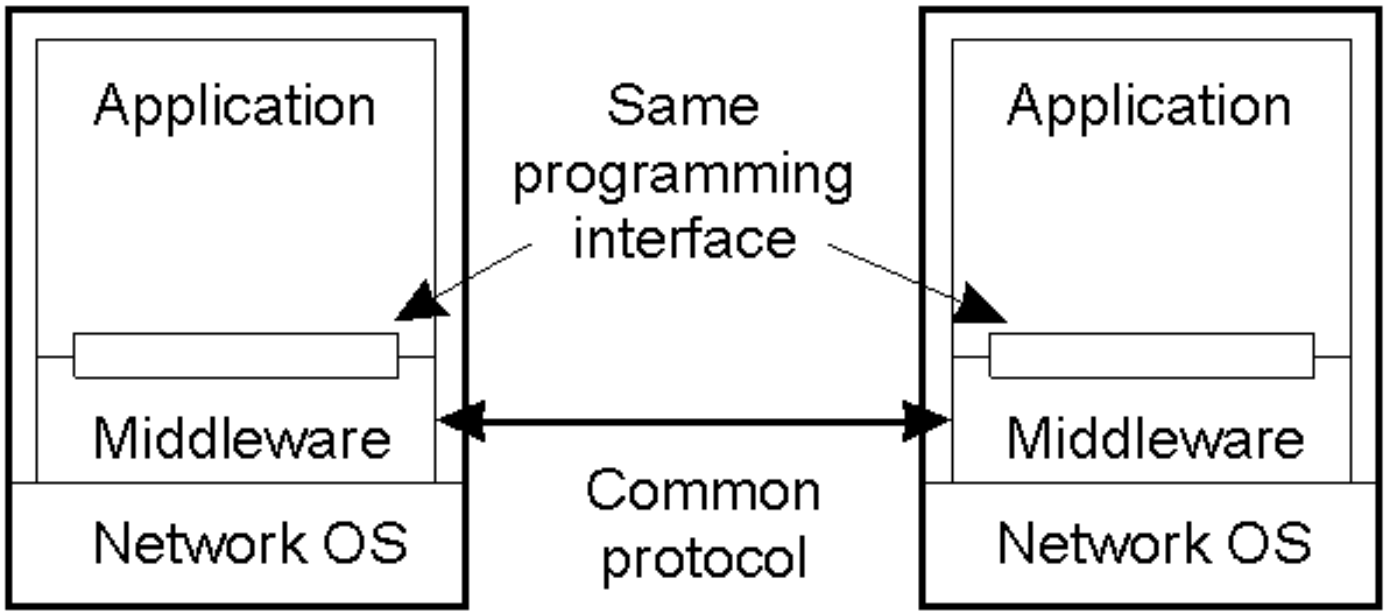
Tanenbaum and van Steen. Distributed Systems: Principles and Paradigms © Prentice Hall 2002

Middleware: Exemplos

- Java RMI
  - Transparência de SO, graças à JVM (Java Virtual Machine)
  - Restrito à programação em Java
- DCOM
  - Transparência de linguagem de programação (LP), mas dependente de SO (Windows)
- CORBA
  - Transparência de SO
  - Transparência de LP

Palavra-chave: Transparência

Middleware e Abertura



Tanenbaum and van Steen. Distributed Systems: Principles and Paradigms © Prentice Hall 2002

- Em um SD baseado em middleware aberto, os protocolos usados por cada camada middleware devem ser os mesmos, assim como as interfaces (de comunicação) que eles apresentam às aplicações

Conceitos de Software

Sistema	Descrição	Principal objetivo
SOD	SO fortemente acoplado para **multiprocessadores e multicomputadores** homogêneos	Esconder e gerenciar recursos de hardware
SOR	SO fracamente acoplado para multicomputadores heterogêneos (LAN e WAN)	Oferecer serviços locais para clientes remotos
Middleware	Camada adicional no topo de SOR que implementa serviços de propósito geral	Fornecer transparência de distribuição

Exemplo: Sistema de Fluxo de Trabalho (Workflow)

- Aplicação: processamento automático de encomendas
- O sistema é usado por várias pessoas de diferentes departamentos, possivelmente em diferentes locais
- Os usuários finais não sabem que as encomendas fluem através do sistema; para eles é como se a operação fosse centralizada

Workflow (cont)

realiza pedido

Depto. de planejamento

pede produto

pede conta

Depto. de estoque

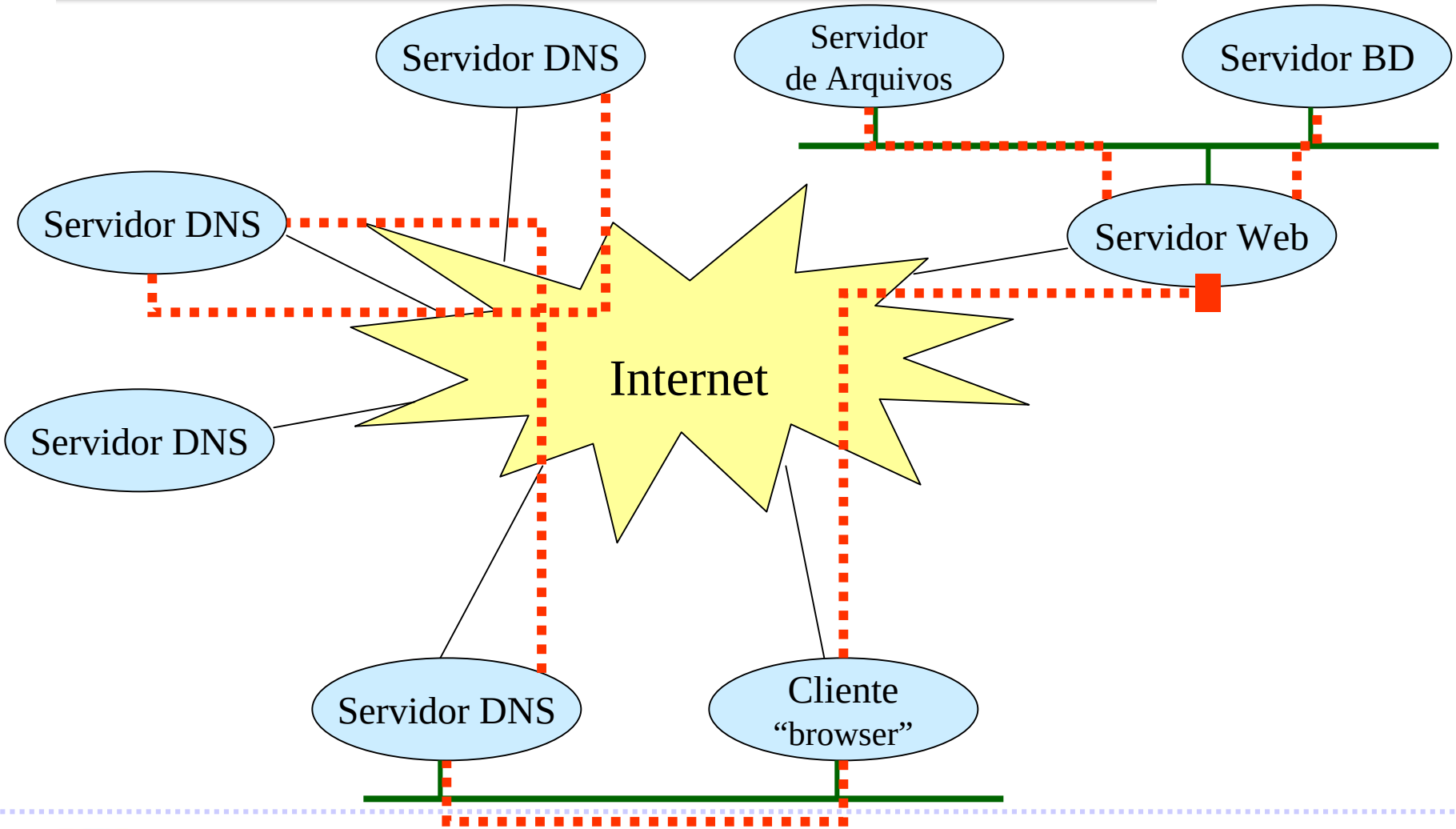
Depto. de contabilidade

sincroniza

envia

Depto. de carga

Web: Um Exemplo de SD?



Web (cont)

- Se a WWW aparecesse para os usuários como um (gigantesco) sistema de documentos centralizado, seria qualificada como um sistema distribuído
- Mas os usuários percebem o fato de que documentos são localizados em lugares diferentes e que são manipulados por diferentes servidores…

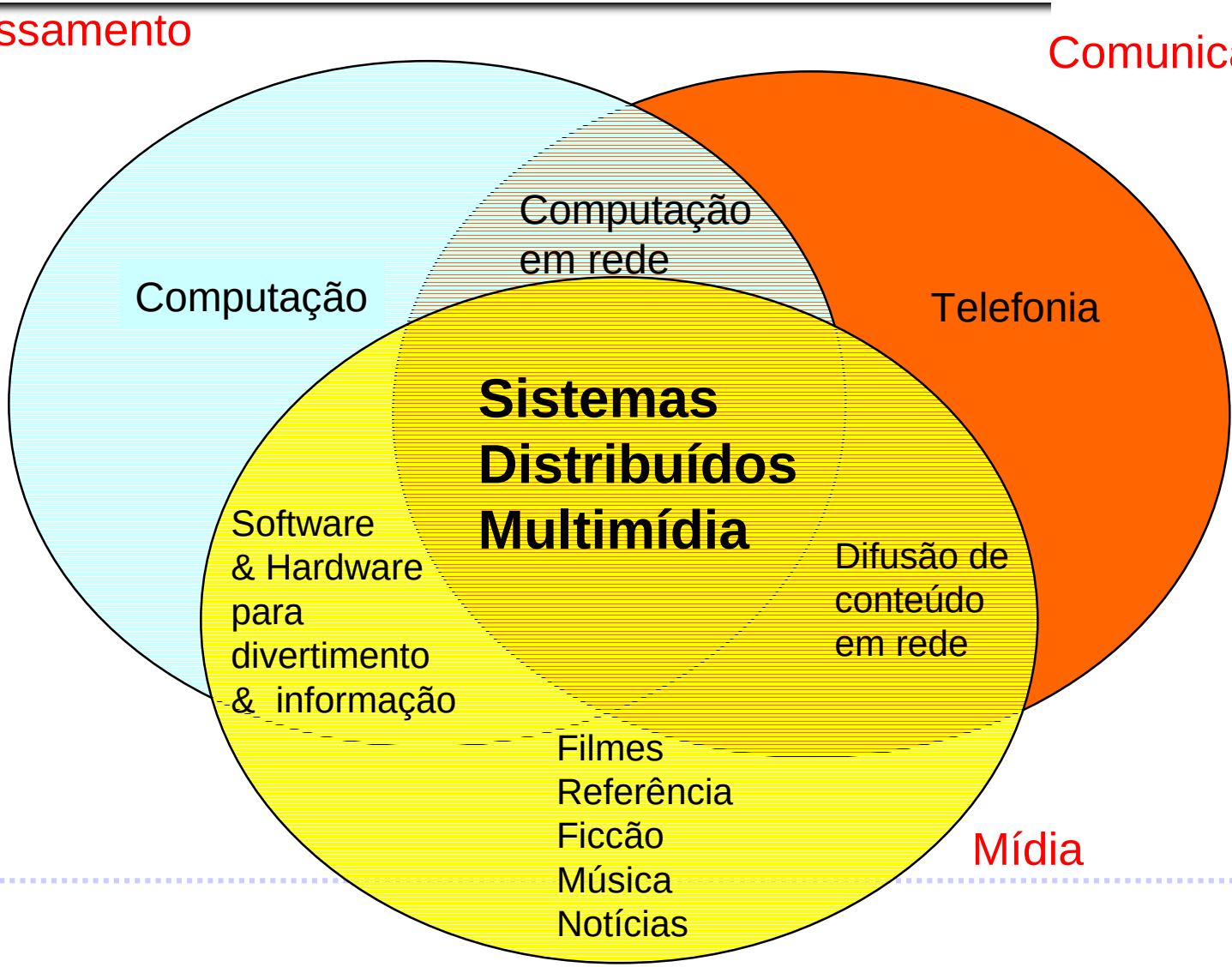
Sistemas Distribuídos: Vantagens

- Economia: melhor relação custo/desempenho
- Eficiência: maior poder total de computação
- Distribuição inerente: máquinas espacialmente separadas
- Confiabilidade: se uma máquina falha, o sistema como um todo pode ainda sobreviver
- Crescimento incremental: poder computacional adicionado em incrementos

Sistemas Distribuídos: Desvantagens

- Pouco software disponível (ainda!)
- *Networking*: a rede pode causar problemas
- Segurança: várias portas de acesso (mas...)

Convergência Digital



Características

- Compartilhamento de recursos
- Abertura
- Concorrência
- Escalabilidade
- Robustez
  - Tolerância a falhas
  - Disponibilidade
- Transparência

Compartilhamento de Recursos (1/3)

- "Recurso" caracteriza o conjunto de elementos que podem ser compartilhados de forma útil em um SD:
  - Hw: impressoras, discos
  - Sw: arquivos, bancos de dados, compiladores
- O compartilhamento reduz custos

Compartilhamento (2/3)

- Recursos fisicamente encapsulados em um dos computadores de um SD só podem ser acessados por outros computadores através de comunicação
- Cada conjunto de recursos de um tipo particular deve ser gerenciado por um programa (Gerenciador de Recursos) que oferece uma interface de comunicação

Compartilhamento (3/3)

- Uma interface de comunicação em um Gerenciador de Recursos permite que os mesmos sejam:
  - acessados
    - deve haver um Esquema de Nomeação para permitir que recursos individuais sejam acessados a partir de qualquer localização
    - mapeamento de nomes de recursos em endereços de comunicação
  - manipulados e atualizados
    - há necessidade de sincronização de acesso concorrente para garantir consistência (Controle de Concorrência)

Abertura

- Determina se um SD pode ser estendido de várias maneiras:
  - extensão por hardware
  - extensão por software

  sem a interrupção ou duplicação de serviços existentes
- Conseguida através da publicação de interfaces, tornando-as disponíveis para desenvolvedores de software
- UNIX é um exemplo de um sistema aberto

Concorrência

- Concorrência e execução paralela existem em um SD por causa de:
  - as atividades separadas de usuários,
  - a independência de recursos e
  - a localização de processos em computadores separados

Problemas de Escalabilidade

Conceito	Exemplo
Serviços centralizados	Um único servidor para todos os usuários
Dados centralizados	Uma única lista telefônica on-line
Algoritmos centralizados	Roteamento baseado em informação completa

Exemplos de limitações de escalabilidade

Escalabilidade

- Filosofia de projeto: se a demanda por um recurso aumentar, deve ser possível estender o sistema para atender à mesma
- SDs devem ser capazes de operar efetiva e eficientemente em escalas diferentes
- Sw de sistema e aplicação não precisam mudar quando a escala do sistema muda
- O processamento deve ser independente do tamanho da rede

Tolerância a Falha

- Em um SD, o hardware essencial para a operação contínua de aplicações críticas pode ser replicado
  - este hw redundante pode ser usado para atividades não-críticas quando não há falhas
- O software pode ser projetado para recuperar o estado de dados permanentes quando uma falha é detectada
- Tipos de falha
  - Transiente: pode acontecer uma vez
  - Intermitente: acontece de tempos em tempos
  - Permanente: acontece sempre

Disponibilidade

- Quando um componente falha em um SD, apenas a parte que usa este componente é afetada; além disso, o componente pode ser re-inicializado em outro computador

⇨ Um SD tem mais partes disponíveis por mais tempo

Transparência

- Esconde do usuário e do programador de aplicação a separação de componentes em um SD
- O sistema é percebido como um todo, em vez de uma coleção de componentes independentes
- Tipos mais comuns de transparência
  - Localização
  - Acesso