

# Computação Distribuída

---

Raquel Mini

[raquelmini@pucminas.br](mailto:raquelmini@pucminas.br)

# CONTEÚDO

---

Introdução à Computação Distribuída

Comunicação entre processos

Sistema de nomes

Sincronização

Exclusão mútua

Eleição de líder

*Deadlock*



1ª PROVA

# CONTEÚDO

---

Objetos distribuídos e invocação remota

Arquivos distribuídos

Análise de complexidade de algoritmos distribuídos

Computação Ubíqua

Redes de Sensores Sem Fio



2ª PROVA

# Sistemas Distribuídos

---

Um sistema distribuído é um conjunto de computadores independentes que se apresenta a seus usuários como um sistema único e coerente

- Os computadores de um sistema distribuído são autônomos
- Para haver um sistema distribuído TEM de haver REDE!
- Visão do usuário: um único sistema integrado

# Sistemas Distribuídos

---

Um sistema distribuído é um conjunto de computadores autônomos interligados em rede e executando programas com o objetivo de disponibilizar recursos aos usuários

- Recursos: componentes de hardware (discos e impressoras), software (arquivos e bancos de dados) e recursos como funcionalidade mais específica (mecanismo de busca)

# Sistemas Distribuídos

---

Um sistema distribuído é aquele no qual os componentes localizados em computadores interligados em rede se comunicam e coordenam suas ações apenas passando mensagens

- Concorrência de componentes
- Falta de um relógio global
- Falhas de componentes independentes

# Sistemas Distribuídos

---

## Exemplos:

- Serviço de Arquivo ou de Impressão
- Aplicação Cliente/Servidor
- WWW (sistema distribuído de larga escala)

## Sistema centralizado: oposto de sistema distribuído

- Exemplo: sistema *mainframe*, acessado via “terminais burros”

# Motivação e Vantagens

---

## Motivação:

- Avanços na tecnologia de redes e de microprocessadores
- Mundo moderno possui aplicações inerentemente distribuídas (ex.: bancos)

## Vantagens (sobre sistemas centralizados):

- Econômica: microprocessadores oferecem uma relação custo/benefício melhor que a de *mainframes* ou computadores de grande porte



# Vantagens

---

## Vantagens (sobre sistemas centralizados):

- Desempenho: um sistema distribuído pode alcançar um desempenho (velocidade, *throughput*) maior que a de um único computador de grande porte
- Algumas aplicações são inerentemente distribuídas e envolvem diversos computadores autônomos

# Vantagens

---

## Vantagens (sobre sistemas centralizados):

- Confiabilidade e tolerância a falhas: se uma máquina trava ou sai da rede, o sistema como um todo pode continuar o processamento
- Escalabilidade: o poder de computação pode ser incrementado aos poucos, independente de arquitetura ou de fabricante

# Desvantagens

---

Complexidade de desenvolvimento

Rede: a rede pode saturar ou causar outros problemas

Segurança: compartilhamento pode gerar acessos não autorizados a dados secretos

# Computação Paralela x Sistemas Distribuídos

---

## Computação Paralela

- Uma aplicação é executada por um conjunto de processadores em um ambiente único (dedicados)
- Comunicação via memória compartilhada
- Objetivo: aumentar *throughput* (vazão) ou reduzir tempo de serviço

## Sistemas Distribuídos

- Descentralizar o sistema
- Compartilhar recursos fisicamente dispersos
- Integrar recursos fisicamente dispersos

# Redes de Computadores X Sistemas Distribuídos X Computação Paralela

<b>Característica</b>	<b>Redes</b>	<b>S. D.</b>	<b>C. P.</b>
Parece um único processador virtual?	Não	Sim	Sim
Todos têm que rodar um mesmo SO?	Não	Não	Sim
Quantas cópias do SO existem	Várias	Várias	1
Como é feita a comunicação?	Arquivos compartilhados	Troca de Mensagens	Memória compartilhada
É necessária padronização de protocolos?	Sim	Sim	Não
Existe apenas uma fila de tarefas?	Não	Não	Sim

# Sistemas Distribuídos

---

Comunicação via troca de mensagens

Troca de mensagens tem custo não desprezível, além de sujeita a falhas e atrasos

Nenhum nó tem conhecimento do estado global do sistema em um dado instante

Objetivo: compartilhar recursos fisicamente separados

# Características

---

## Heterogeneidade

- A Internet permite aos usuários acessarem serviços e executarem aplicativos por meio de um conjunto heterogêneo de computadores e redes
- A heterogeneidade se aplica aos seguintes aspectos:
  - Redes
  - Hardware de computador
  - Sistemas operacionais
  - Linguagem de programação
  - Implementações de diferentes desenvolvedores

# Características

---

Transparência no compartilhamento de recursos:  
“esconder” dos usuários a distribuição física dos recursos

- Acesso: recursos locais e remotos acessados igualmente
- Localização: usuários não sabem a localização dos recursos
- Migração: recursos podem se mover sem trocar de nomes
- Replicação: usuários não sabem o nº de cópias de um recurso
- Concorrência: vários usuários compartilham recursos automaticamente
- Paralelismo: aplicações podem rodar em paralelo sem o conhecimento do usuário



# Características

---

## Escalabilidade

- Os sistemas distribuídos funcionam de forma efetiva e eficaz em muitas escalas diferentes
- Aplicações não necessitam de mudanças quando o tamanho do sistema aumenta
- Um sistema é escalável se permanece eficiente quando há um aumento significativo no número de recursos e no número de usuários

# Características

---

## Tolerância a falhas

- Capacidade de um sistema detectar uma falha e então proceder de uma das seguintes formas:
  - Terminar corretamente
  - Contornar a falha (usuários não conseguem percebê-la)
- Disponibilidade: fração de tempo que o sistema está disponível

# Características

---

## Sistemas abertos

- Capacidade de ser estendido e reimplementado de várias maneiras
- Importante: interfaces públicas e aderentes a padrões
- Os sistemas projetados a partir de padrões públicos são chamados de sistemas distribuídos abertos

# Tendências em Sistemas Distribuídos

---

O surgimento da tecnologia de redes pervasivas

O surgimento da computação ubíqua, combinado ao desejo de suportar mobilidade do usuário em sistemas distribuídos

A crescente demanda por serviços multimídia

A visão dos sistemas distribuídos como um serviço público

# Modelos de Sistema

---

Não existe a noção de relógio global em um sistema distribuído

Os relógios de diferentes computadores não fornecem necessariamente a mesma hora

Toda a comunicação entre processos é obtida por meio de troca de mensagens

# Arquitetura para Sistemas Distribuídos

---

A arquitetura de um sistema é sua estrutura em termos de componentes especificados separadamente e suas inter-relações

O objetivo global é garantir que a estrutura atenda às demandas atuais e, provavelmente, às futuras demandas impostas sobre ela

# Arquitetura para Sistemas Distribuídos

---

## Elementos arquitetônicos

- Quais são as **entidades** que estão se comunicando no sistema distribuído?
- Como elas se **comunicam** ou, mais especificamente, qual é o paradigma de comunicação utilizado?
- Quais **funções** e responsabilidades (possivelmente variáveis) estão relacionadas a eles na arquitetura global?
- Como eles são **mapeados** na infraestrutura distribuída física (qual é sua localização)?

# Entidades

---

As entidades que se comunicam em um sistema distribuído normalmente são processos

Um sistema distribuído é composto de processos acoplados a paradigmas de comunicação apropriados entre processos



# Comunicação

---

Existem três tipos de paradigma de comunicação

- Comunicação entre processos
- Invocação remota
- Comunicação indireta

Na comunicação entre processos e na invocação remota, a comunicação representa uma relação bilateral entre um remetente e um destinatário

# Funções

---

Em um sistema distribuído, os processos, componentes ou serviços interagem uns com os outros para realizar uma atividade útil, por exemplo, para suportar uma sessão de bate-papo

Ao fazer isso, os processos assumem determinadas funções

Essa função é fundamental no estabelecimento da arquitetura global a ser adotada

- Cliente-servidor
- Peer-to-peer

# Arquitetura Cliente-Servidor

---

Os processos assumem os papéis de clientes ou servidores

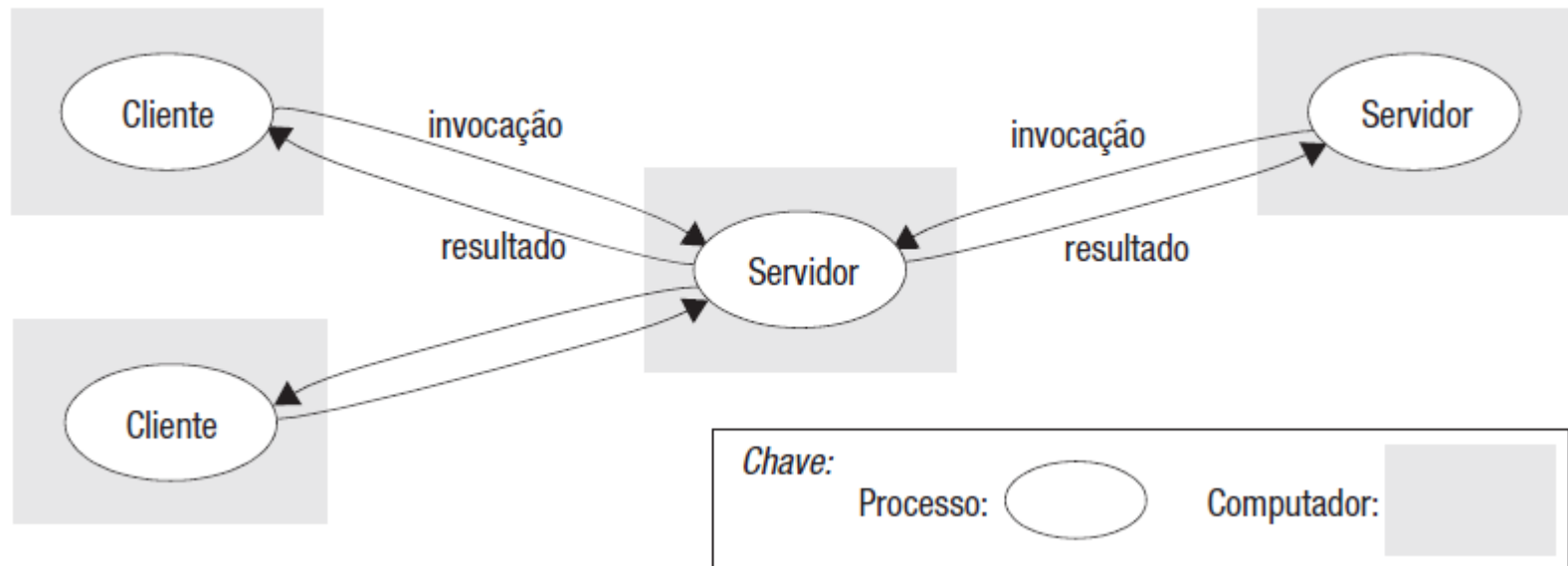
Os processos clientes interagem com processos servidores, localizados possivelmente em distintos computadores hospedeiros, para acessar os recursos compartilhados que estes gerenciam

Os servidores podem ser clientes de outros servidores

- Exemplo: Os servidores Web são clientes do serviço DNS

# Arquitetura Cliente-Servidor

Os clientes chamam o servidor individual



# Arquitetura Peer-to-Peer

---

Todos os processos envolvidos em uma tarefa desempenham funções semelhantes, interagindo cooperativamente como pares (peers) sem distinção entre processos clientes e servidores

Todos os processos participantes executam o mesmo programa e oferecem o mesmo conjunto de interfaces uns para os outros

Modelo peer-to-peer é mais flexível em termos de escalabilidade do que o modelo cliente-servidor

# Arquitetura Peer-to-Peer

---

O objetivo é explorar os recursos de um grande número de computadores para o cumprimento de uma dada tarefa ou atividade

- Exemplo: Napster, BitTorrent

# Arquitetura Peer-to-Peer

---

## Funcionamento:

- Grande número de processos executados em diferentes computadores
- Grande número de objetos de dados são compartilhados
- Um computador individual contém apenas uma pequena parte do banco de dados do aplicativo
- As cargas são distribuídas por muitos computadores e conexões de rede
- Cada objeto é replicado em vários computadores para distribuir a carga

# Arquitetura Peer-to-Peer

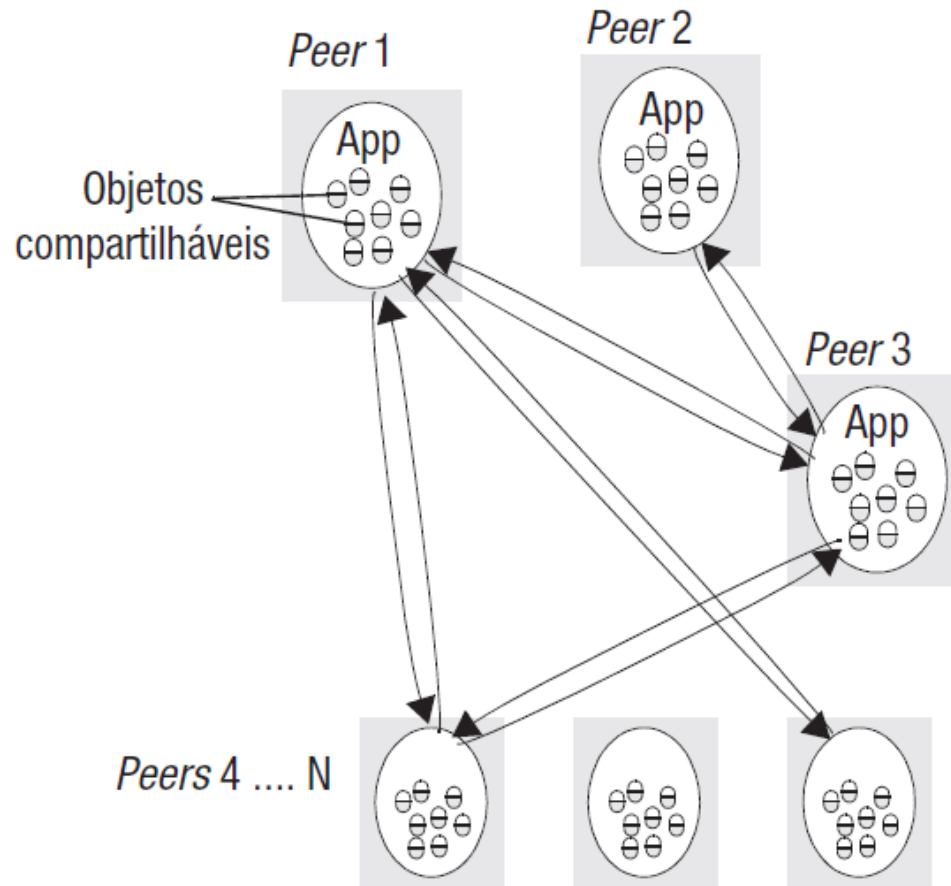
---

A necessidade de colocar objetos individuais, recuperá-los e manter réplicas entre muitos computadores torna essa arquitetura mais complexa do que a arquitetura cliente-servidor



# Arquitetura Peer-to-Peer

---



# Mapeamento

---

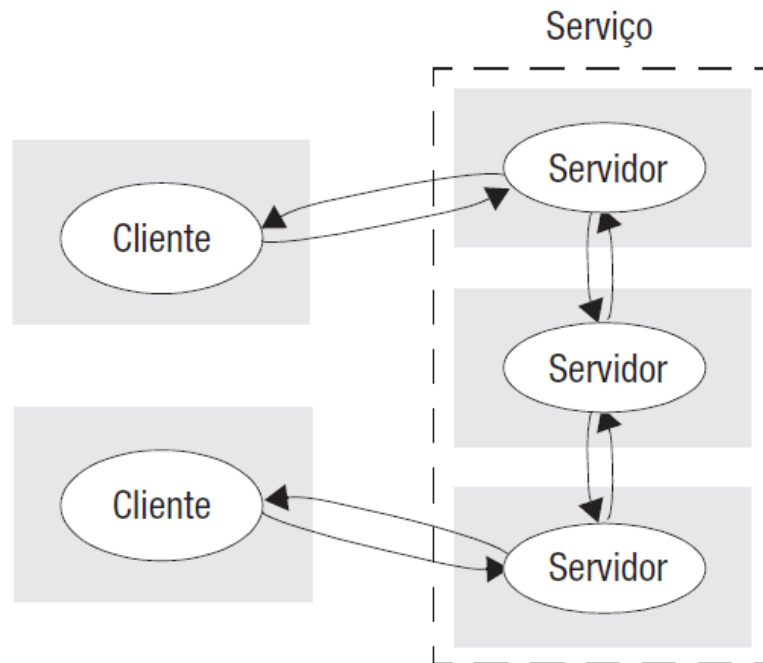
De que modo entidades como objetos ou serviços são mapeados na infraestrutura física distribuída subjacente

Onde colocar determinado cliente ou servidor?

- Mapeamento de serviços em vários servidores
- Uso de cache
- Código móvel

# Mapeamento de serviços em vários servidores

Vários processos servidores em diferentes computadores hospedeiros, interagindo conforme necessário para fornecer um serviço para processos clientes



# Uso de cache

---

Realizar um armazenamento de objetos de dados recentemente usados em um local mais próximo a um cliente do que a origem real dos objetos em si

Quando um processo cliente requisita um objeto, o serviço de cache primeiro verifica se possui armazenado uma cópia atualizada desse objeto

Caso esteja disponível, ele é entregue ao processo cliente

# Uso de cache

---

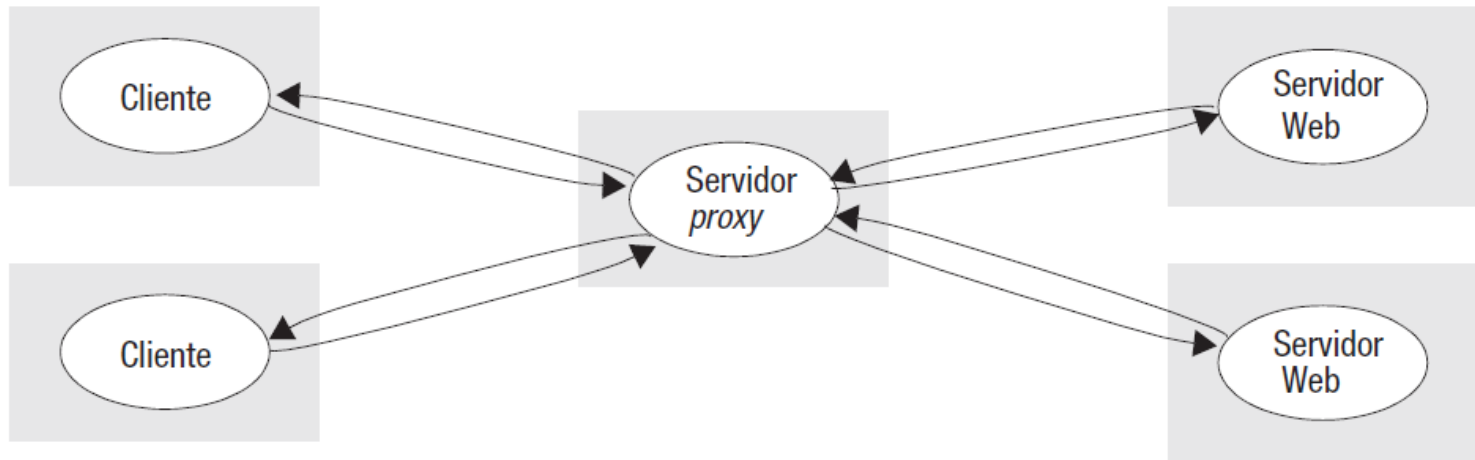
Se o objeto não estiver armazenado, ou se a cópia não estiver disponível, ele é acessado diretamente em sua origem

Um servidor proxy Web fornece uma cache compartilhada de recursos Web para máquinas clientes de um ou vários sites

# Uso de cache

---

## Serviço proxy Web



# Código Móvel

---

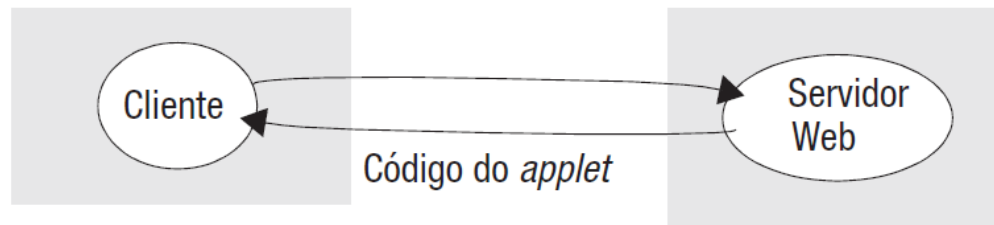
Código de programa que pode ser transferido de um computador para outro e ser executado no destino

- Exemplo: *applets* Java

# Código Móvel

---

Requisição do cliente resulta no *download* do código de um *applet*



O cliente interage com o *applet*





# Tarefa 1 – postar no Canvas até 21/02/2021

---

Use a World Wide Web (WWW) como exemplo para ilustrar o conceito de compartilhamento de recursos, cliente e servidor.

- Apresente a história da WWW.
- Descreva o funcionamento básico da WWW.
- Quais são as vantagens e desvantagens das tecnologias básicas HTML, URLs e HTTP para navegação em informações? Alguma dessas tecnologias é conveniente como base para a computação cliente-servidor em geral?