

# Camada de Aplicação

---

Introdução a Redes  
Prof. Raquel Cruz





# Camada de Aplicação

- Os protocolos de rede são projetados para suportar as diversas aplicações de rede que conhecemos:
  - Correio Eletrônico
  - Transferências de arquivo
  - Grupos de discussão
  - Bate-papo
  - Videoconferência
  - Telefonia por Internet

# Camada de Aplicação (cont.)



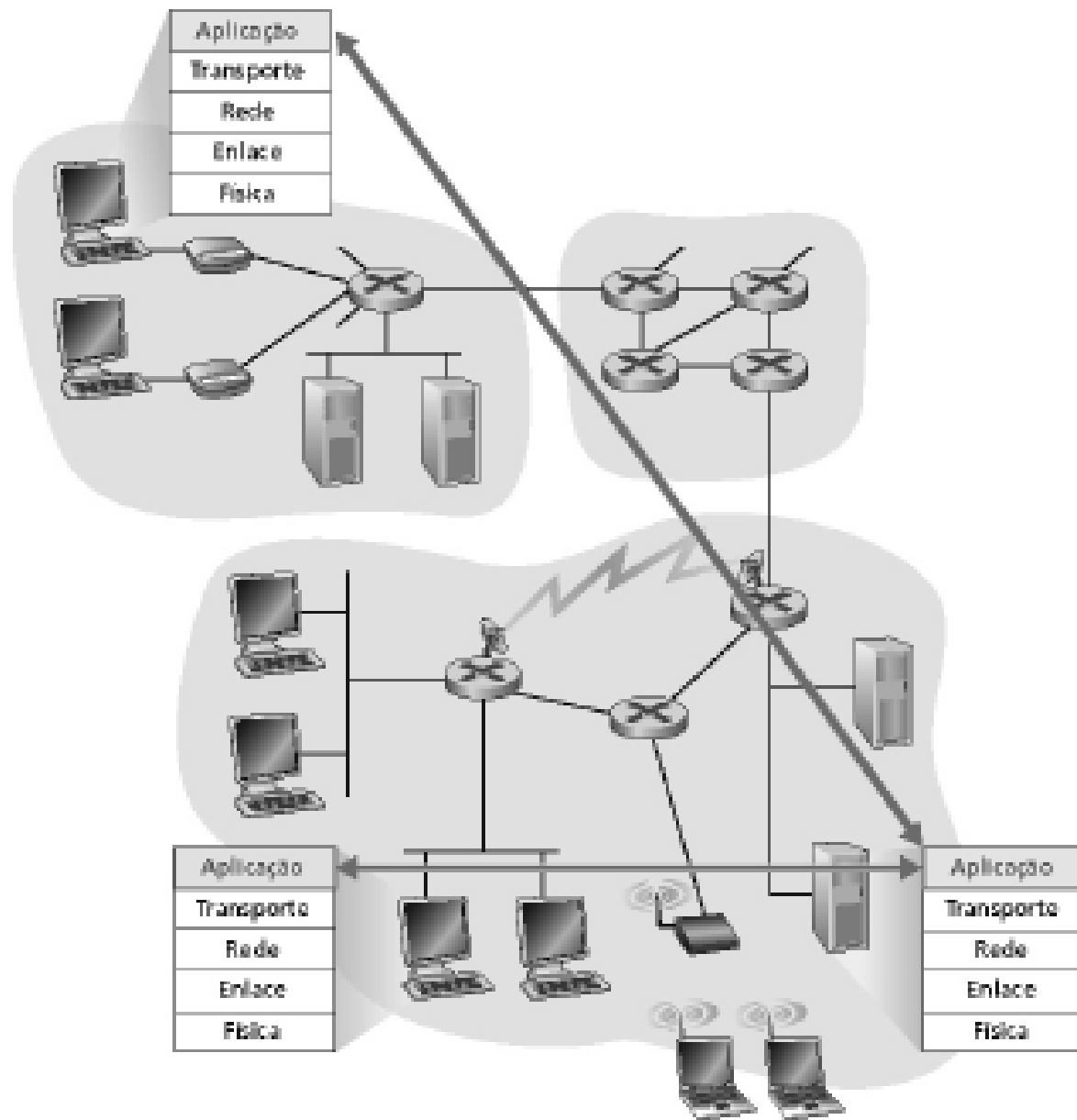
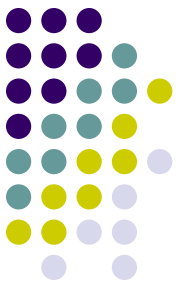
- Um software de aplicação de rede é distribuído entre dois ou mais sistemas finais
- Na aplicação Web, por exemplo, a comunicação ocorre entre o browser no host e o software no servidor Web
- Esses softwares na verdade são **processos** que se comunicam através da troca de mensagens por meio da rede de computadores



## Camada de Aplicação (cont.)

- O processo de origem cria e envia mensagens para a rede
- O processo de destino recebe essas mensagens e possivelmente responde, devolvendo outras mensagens
- As aplicações de rede têm protocolos de camada de aplicação que definem tanto o formato e a ordem das mensagens trocadas entre os processos como as ações realizadas na transmissão e na recepção de uma mensagem

# Aplicações comunicantes





# Camada de Aplicação (cont.)

- Aplicações de rede X protocolos de camada de aplicação
- Exemplo:
  - A Web é uma aplicação de rede que permite que os usuários obtenham “documentos” de servidores Web por demanda
  - Nessa aplicação há diversos componentes: padrão para formato de documentos (HTML); browsers (Firefox, Internet Explorer, Chrome, etc); servidores Web (Apache, IIS); protocolo da camada de aplicação
  - O protocolo utilizado pela Web, o HTTP, define como as mensagens são passadas entre o browser e o servidor Web

# Camada de Aplicação (cont.)



- Um protocolo de camada de aplicação define como os processos de aplicação, que rodam em diferentes sistemas finais, passam mensagens uns para os outros. Um protocolo de camada de aplicação define:
  - Os tipos de mensagens trocadas
  - A sintaxe dos vários tipos de mensagens
  - A semântica dos campos
  - As regras

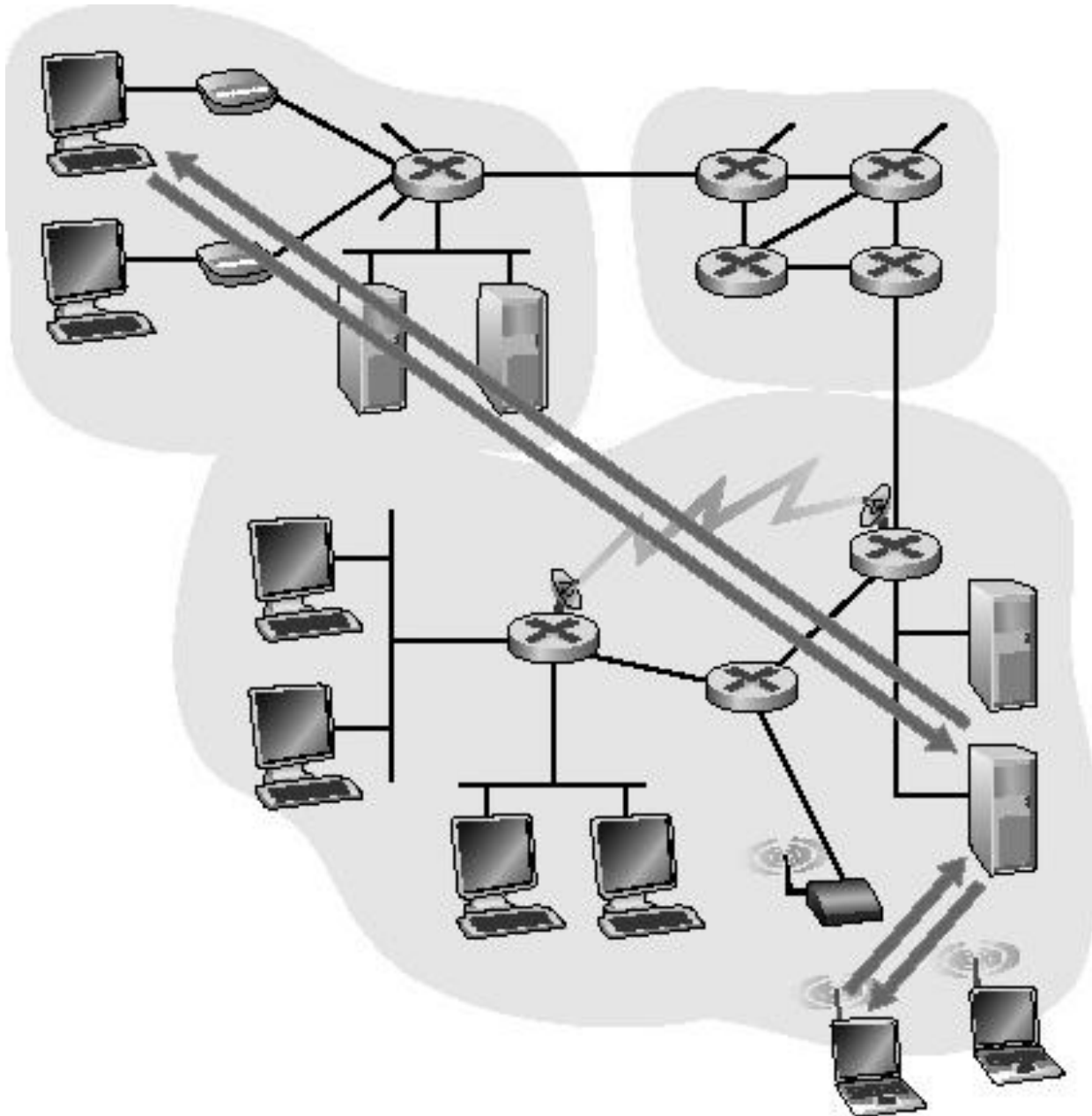


# Clientes e Servidores

- Um protocolo de aplicação de rede tem normalmente duas partes: o lado **cliente** e o lado **servidor**
- Exemplo:
  - Um browser Web implementa o lado cliente do HTTP e um servidor Web implementa o lado servidor do HTTP



# Interação Cliente-Servidor

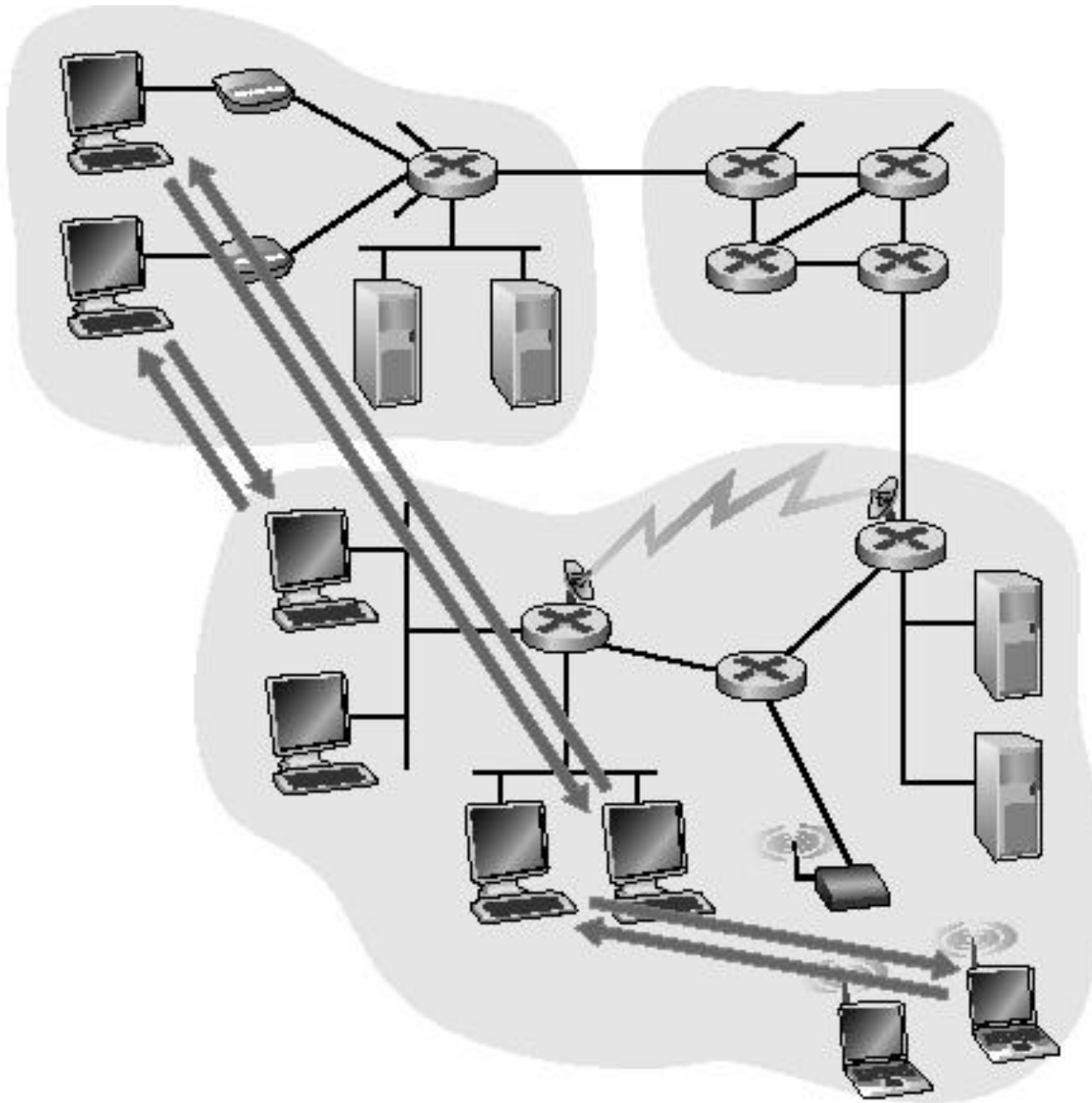


# Arquitetura P2P

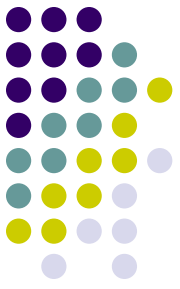


- No modelo *peer-to-peer* (par-a-par) não há um servidor sempre funcionando no centro da aplicação
- Nenhum dos *hosts* participantes precisa estar sempre em funcionamento
- É possível que o *host* mude de endereço IP cada vez que é ligado

# Aplicação P2P

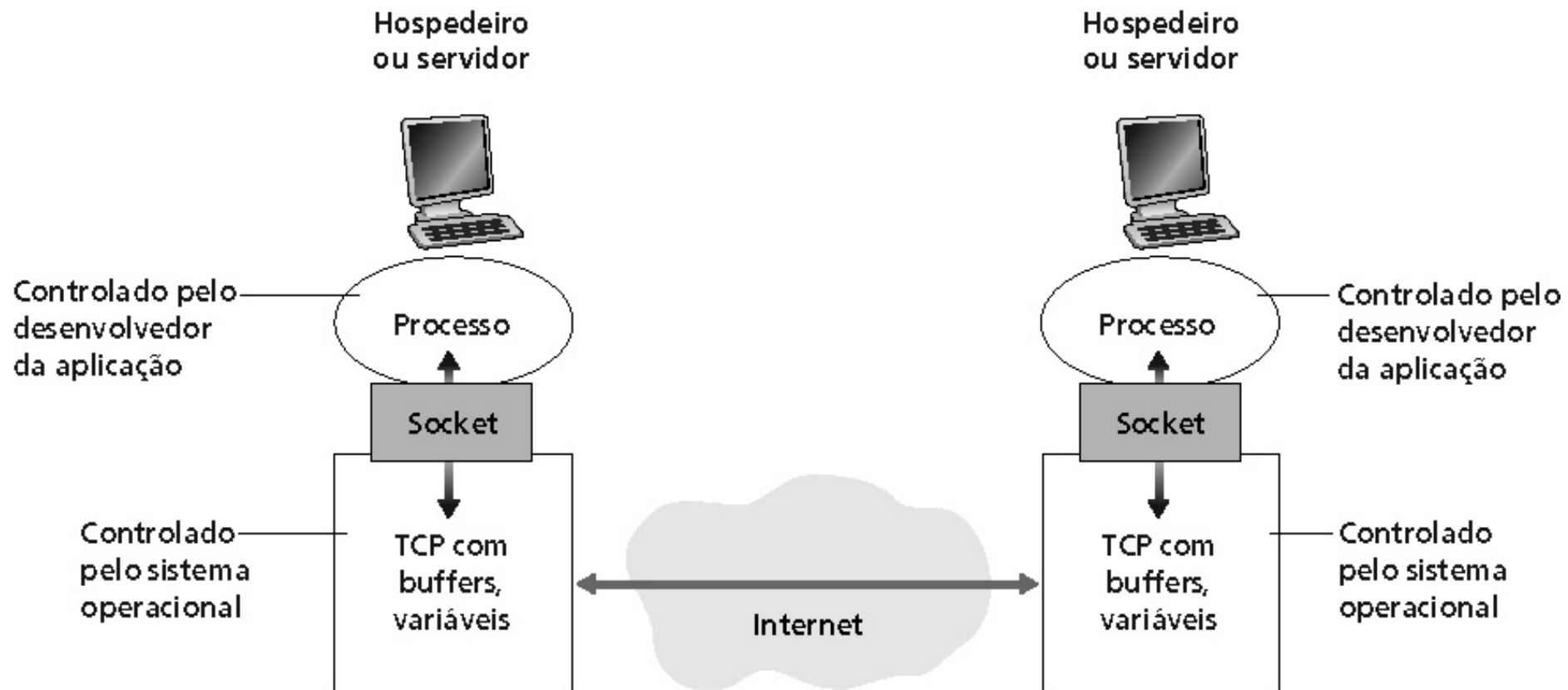
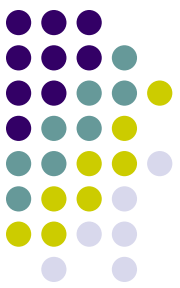


# Processos que se comunicam por meio de uma rede



- Uma aplicação envolve dois processos em *hosts* diferentes que se comunicam
- Processo: programa executando em um *host*
- Os dois processos se comunicam enviando e recebendo mensagens através de suas **portas**
- Processo cliente: processo que inicia a comunicação
- Processo servidor: processo que espera para ser contatado

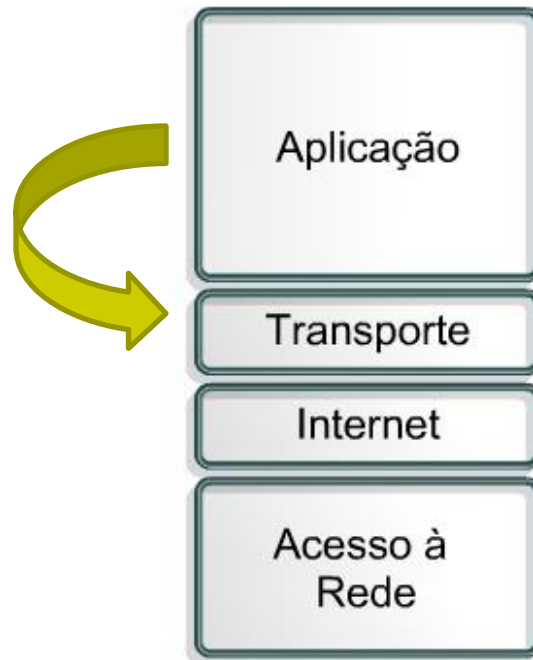
# Processos de aplicação



# Processos que se comunicam por meio de uma rede (cont.)



- Uma porta ou *socket* é a interface entre a camada de aplicação e a camada de transporte



# Endereçamento de processos



- Para um processo receber mensagens, ele deve ter um identificador
- Um *host* possui um endereço IP de 32 bits
- O endereço IP do *host* onde o processo está executando é suficiente para identificar o processo?
- Não, muitos processos podem estar em execução no mesmo *host*
- O identificador inclui o endereço IP e o **número da porta**

# De que serviços de transporte uma aplicação necessita?



- Perda de dados
  - Algumas aplicações (ex.: áudio) podem tolerar alguma perda
  - Outras aplicações (ex.: transferência de arquivos, telnet) exigem transferência de dados 100% confiável

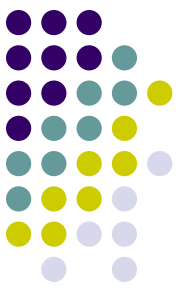


# De que serviços de transporte uma aplicação necessita? (cont.)



- Largura de banda
  - Algumas aplicações (ex.: multimídia) exigem uma banda mínima para serem “efetivas”
  - Outras aplicações (“aplicações elásticas”) melhoram quando a banda disponível aumenta
- Temporização
  - Algumas aplicações (ex.: telefonia Internet, jogos interativos) exigem baixos atrasos para serem “efetivos”

# Requisitos de transporte de aplicações comuns



Aplicação	Perdas	Banda	Sensível ao atraso
file transfer	sem perdas	elástica	não
e-mail	sem perdas	elástica	não
Web documents	tolerante	elástica	não
real-time áudio/vídeo	tolerante	áudio: 5 Kb-1 Mb vídeo: 10 Kb-5 Mb	sim, 100's mseg
stored áudio/video	tolerante	igual à anterior	sim, segundos
jogos interativos	tolerante	kbps	sim, 100's mseg
e-business	sem perda	elástica	sim

# Serviços dos protocolos de transporte da Internet



- Serviço TCP
  - Orientado à conexão: conexão requerida entre processos cliente e servidor
  - Transporte confiável entre os processos de envio e recepção
  - Controle de fluxo: o transmissor não sobrecarrega o receptor
  - Controle de congestionamento: protege a rede do excesso de tráfego
  - Não oferece: garantias de temporização e de banda mínima

# Serviços dos protocolos de transporte da Internet (cont.)



- Serviços UDP
  - Transferência de dados não confiável entre os processos transmissor e receptor
  - Não oferece: estabelecimento de conexão, confiabilidade, controle de fluxo e de congestionamento, garantia de temporização e de banda mínima
- Porque ambos? Porque existe o UDP?

# Referência



KUROSE, James F. Redes de computadores e a internet, Editora: Addison Wesley. 2003.