

# Capítulo 2

## Camada de aplicação

### Nota sobre o uso destes slides ppt:

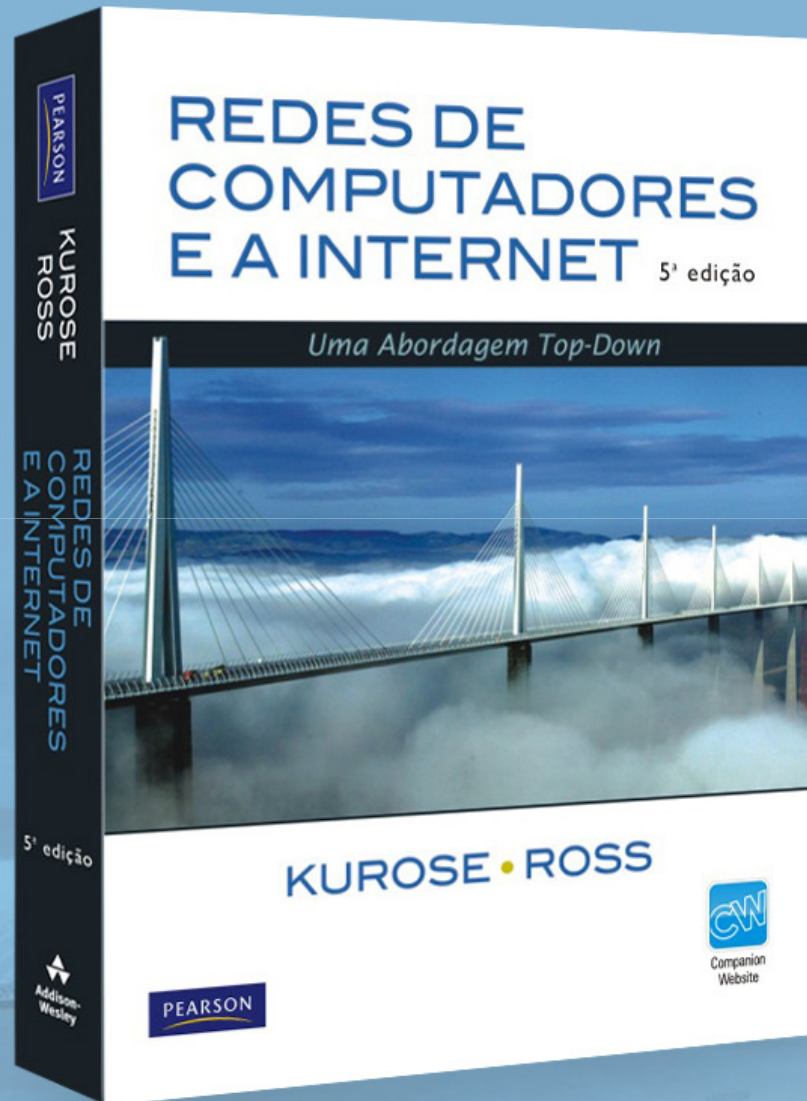
Estamos disponibilizando estes slides gratuitamente a todos (professores, alunos, leitores). Eles estão em formato do PowerPoint para que você possa incluir, modificar e excluir slides (incluindo este) e o conteúdo do slide, de acordo com suas necessidades. Eles obviamente representam *muito* trabalho da nossa parte. Em retorno pelo uso, pedimos apenas o seguinte:

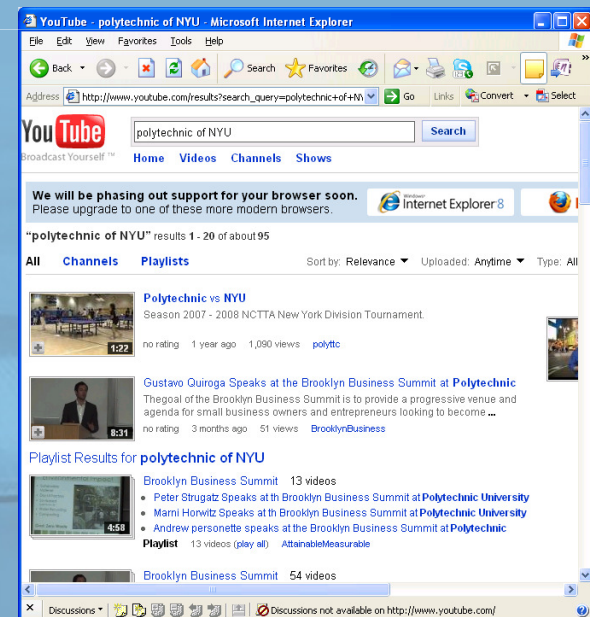
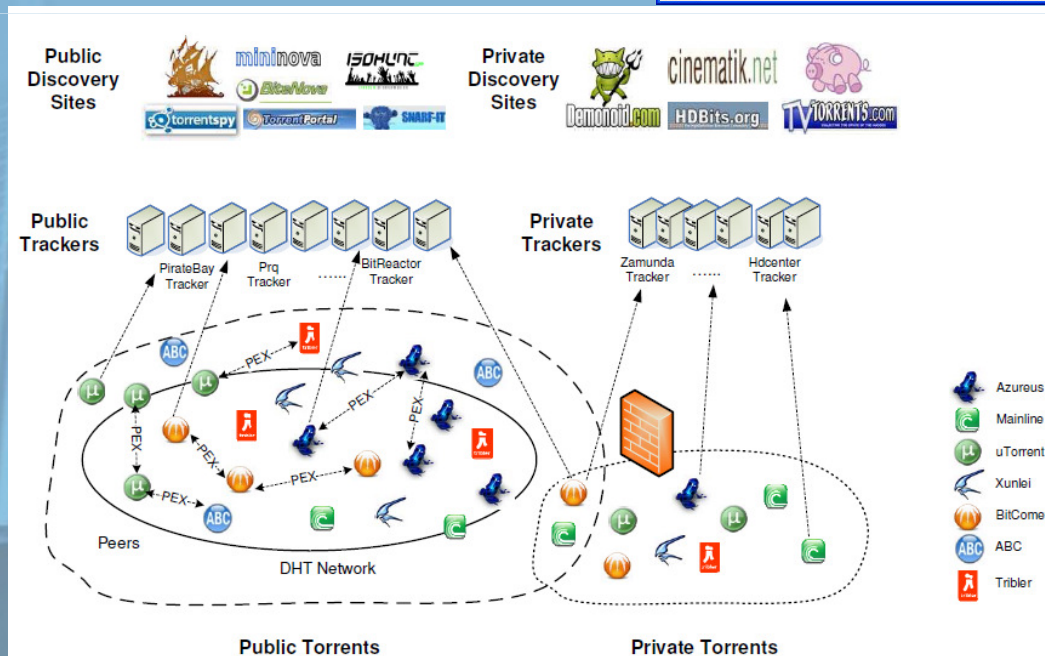
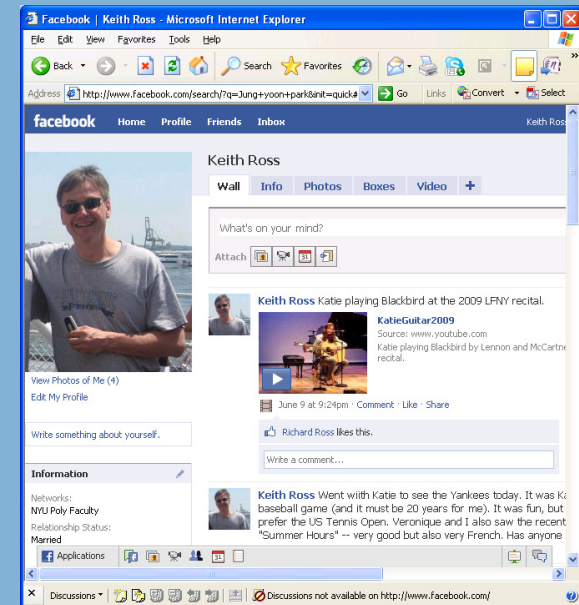
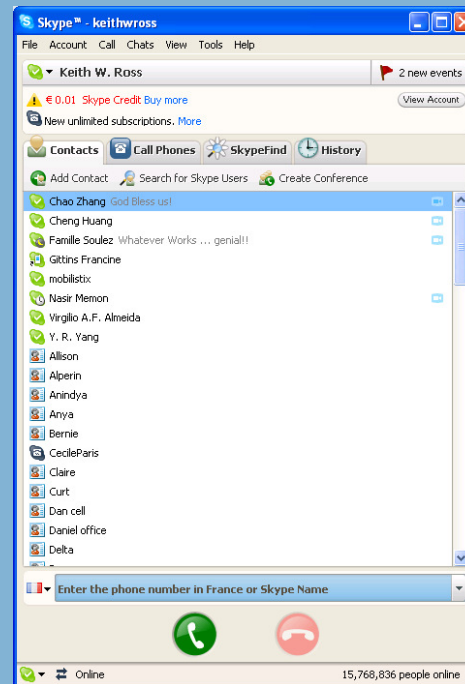
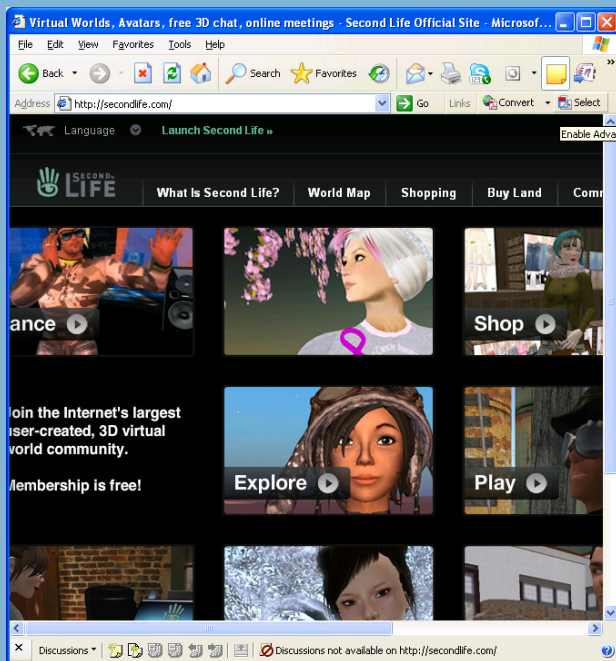
- ❑ Se você usar estes slides (por exemplo, em sala de aula) sem muita alteração, que mencione sua fonte (afinal, gostamos que as pessoas usem nosso livro!).
- ❑ Se você postar quaisquer slides sem muita alteração em um site Web, que informe que eles foram adaptados dos (ou talvez idênticos aos) nossos slides, e inclua nossa nota de direito autoral desse material.

Obrigado e divirta-se! JFK/KWR

Todo o material copyright 1996-2009

J. F Kurose e K. W. Ross, Todos os direitos reservados.





# Capítulo 2: Camada de aplicação

REDES DE  
COMPUTADORES  
E A INTERNET 5ª edição

*Uma Abordagem Top-Down*

- ❑ 2.1 Princípios de aplicações de rede
- ❑ 2.2 A Web e o HTTP
- ❑ 2.3 FTP
- ❑ 2.4 Correio eletrônico
  - ❖ SMTP, POP3, IMAP
- ❑ 2.5 DNS
- ❑ 2.6 Aplicações P2P
- ❑ 2.7 Programação de sockets com UDP
- ❑ 2.8 Programação de sockets com TCP



# Capítulo 2: Camada de aplicação

REDES DE  
COMPUTADORES  
E A INTERNET 5ª edição

Uma Abordagem Top-Down

## Objetivos do capítulo:

- ❑ aspectos conceituais, de implementação de protocolos de aplicação de rede
  - ❖ modelos de serviço da camada de transporte
  - ❖ paradigma cliente-servidor
  - ❖ paradigma *peer-to-peer*
- ❑ aprenda sobre protocolos examinando protocolos populares em nível de aplicação
  - ❖ HTTP
  - ❖ FTP
  - ❖ SMTP/POP3/IMAP
  - ❖ DNS
- ❑ programando aplicações de rede
  - ❖ API socket

# Algumas aplicações de rede

- ☐ e-mail
- ☐ web
- ☐ mensagem instantânea
- ☐ login remoto
- ☐ compartilhamento de arquivos P2P
- ☐ jogos em rede multiusuários
- ☐ clipes de vídeo armazenados em fluxo contínuo
- ☐ redes sociais
- ☐ voice over IP
- ☐ vídeoconferência em tempo real
- ☐ computação em grade

# Criando uma aplicação de rede

## Escreva programas que

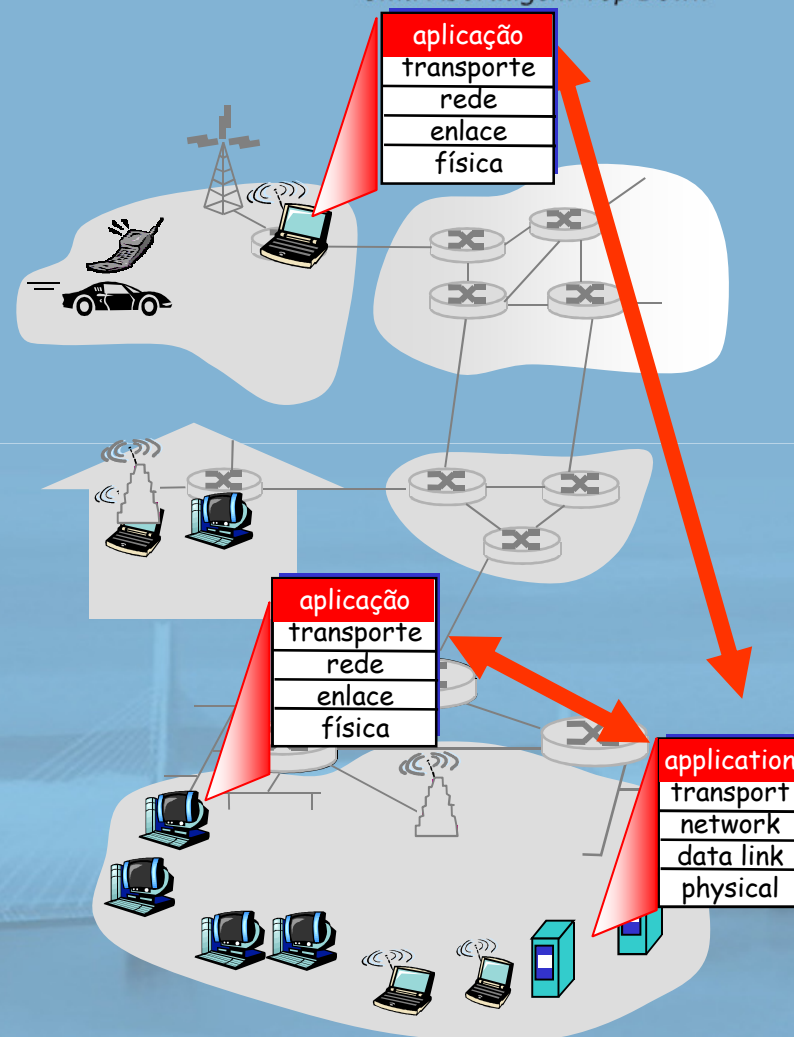
- ❖ executem em (diferentes) *sistemas finais*
- ❖ se comuniquem pela rede
- ❖ p. e., software de servidor Web se comunica com software de navegador Web

## Não é preciso escrever software para dispositivos do núcleo da rede

- ❖ dispositivos do núcleo da rede não executam aplicações do usuário
- ❖ as aplicações nos sistemas finais permitem rápido desenvolvimento e propagação

## REDES DE COMPUTADORES E A INTERNET 5ª edição

Uma Abordagem Top-Down



# Capítulo 2: Camada de aplicação

REDES DE  
COMPUTADORES  
E A INTERNET 5ª edição

*Uma Abordagem Top-Down*

- ❑ 2.1 Princípios de aplicações de rede
- ❑ 2.2 A Web e o HTTP
- ❑ 2.3 FTP
- ❑ 2.4 Correio eletrônico
  - ❖ SMTP, POP3, IMAP
- ❑ 2.5 DNS
- ❑ 2.6 Aplicações P2P
- ❑ 2.7 Programação de sockets com UDP
- ❑ 2.8 Programação de sockets com TCP

# Arquiteturas de aplicação

REDES DE  
COMPUTADORES  
E A INTERNET 5ª edição

*Uma Abordagem Top-Down*

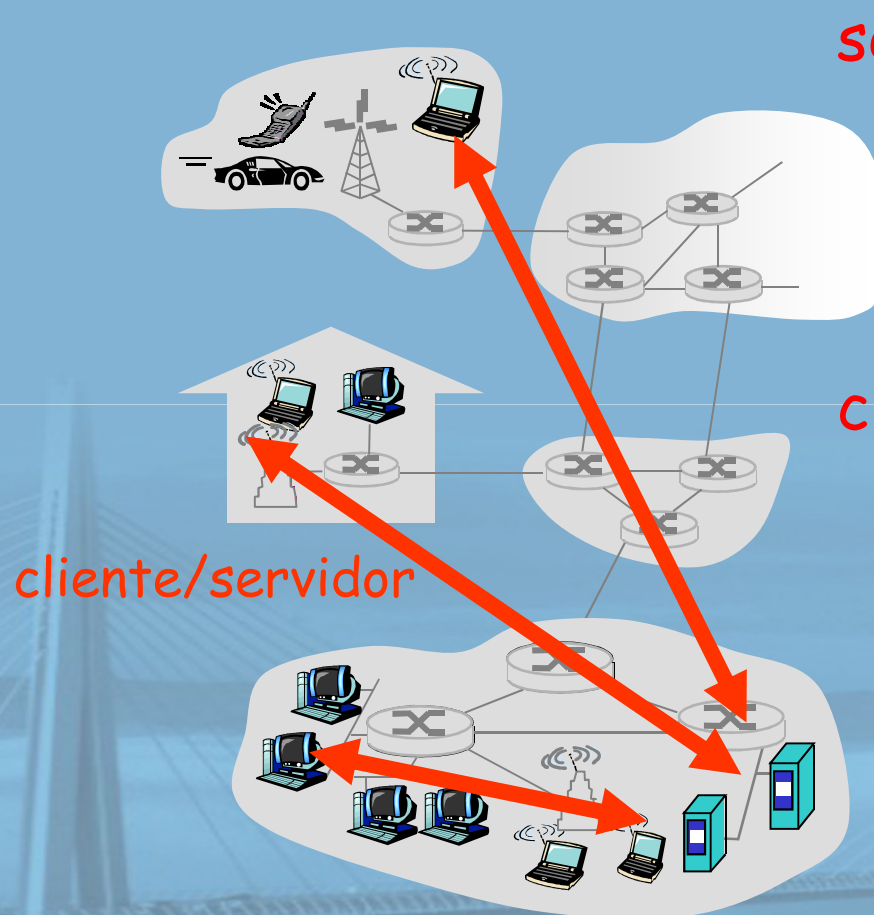
- ❑ Cliente-servidor
  - ❖ Incluindo centros de dados/cloud computing
- ❑ Peer-to-peer (P2P)
- ❑ Híbrida de cliente-servidor e P2P



# Arquitetura cliente-servidor

REDES DE  
COMPUTADORES  
E A INTERNET 5ª edição

*Uma Abordagem Top-Down*



## servidor:

- ❖ hospedeiro sempre ligado
- ❖ endereço IP permanente
- ❖ server farms por expansão

## clientes:

- ❖ comunicam-se com o servidor
- ❖ podem estar conectados intermitentemente
- ❖ podem ter endereços IP dinâmicos
- ❖ não se comunicam diretamente entre si

# Centros de dados da Google

REDES DE  
COMPUTADORES  
E A INTERNET 5ª edição

*Uma Abordagem Top-Down*

- ❑ custo estimado do centro de dados: \$600M
- ❑ Google gastou \$2,4B em 2007 em novos centros de dados
- ❑ cada centro de dados usa de 50 a 100 megawatts de potência



# Arquitetura P2P pura

- ❑ *nenhum* servidor sempre ligado
- ❑ sistemas finais arbitrários se comunicam diretamente
- ❑ pares são conectados intermitentemente e mudam endereços IP



altamente escalável, mas  
difícil de administrar



# Híbrido de cliente-servidor e P2P

## Skype

- ❖ aplicação P2P voice-over-IP P2P
- ❖ servidor centralizado: achando endereço da parte remota:
- ❖ conexão cliente-cliente: direta (não através de servidor)

## Mensagem instantânea

- ❖ bate-papo entre dois usuários é P2P
- ❖ serviço centralizado: detecção/localização da presença do cliente
  - usuário registra seu endereço IP com servidor central quando entra on-line
  - usuário contacta servidor central para descobrir endereços IP dos parceiros



# Processos se comunicando

- processo:** programa rodando dentro de um hospedeiro
- ❑ no mesmo hospedeiro, dois processos se comunicam usando a **comunicação entre processos** (definida pelo SO).
  - ❑ processos em hospedeiros diferentes se comunicam trocando **mensagens**

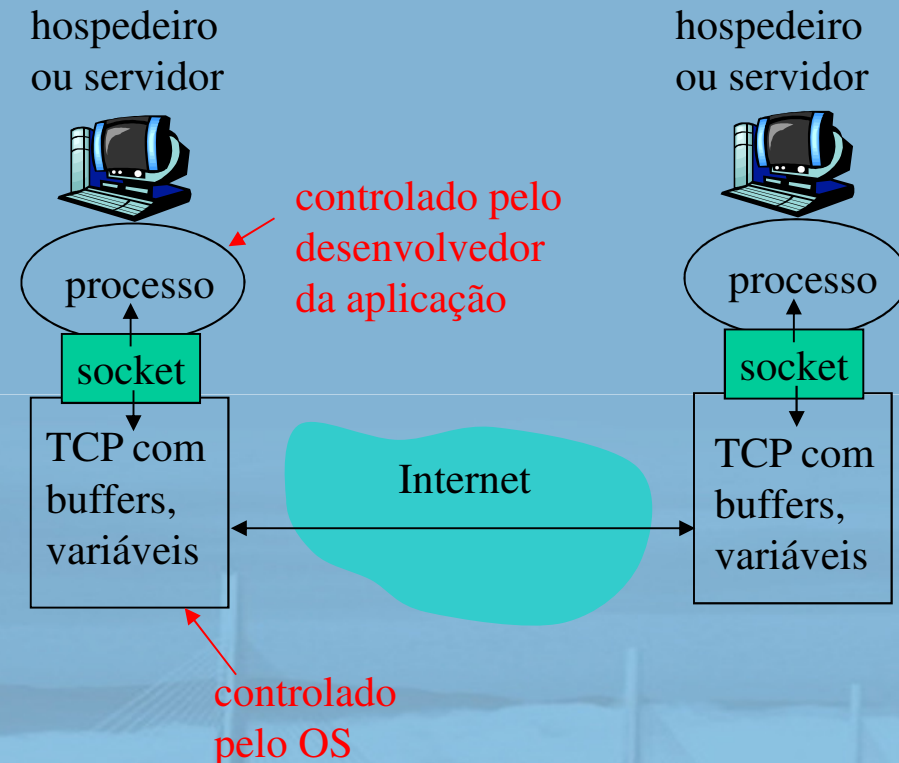
**processo cliente:**  
processo que inicia a comunicação

**processo servidor:**  
processo que espera para ser contactado

- ❑ Nota: aplicações com arquiteturas P2P têm processos clientes & processos servidores

# Sockets

- ❑ processo envia/recebe mensagens de/para seu **socket**
- ❑ socket semelhante à porta
  - ❖ processo enviando empurra mensagem pela porta
  - ❖ processo enviando conta com infraestrutura de transporte no outro lado da porta, que leva a mensagem ao socket no processo receptor
- ❑ API: (1) escolha do protocolo de transporte; (2) capacidade de consertar alguns parâmetros **(muito mais sobre isso adiante)**



# Endereçando processos

- ❑ para receber mensagens, processo deve ter *identificador*
- ❑ dispositivo hospedeiro tem endereço IP exclusivo de 32 bits
- ❑ exercício: use `ipconfig` do comando prompt para obter seu endereço IP (Windows)
- ❑ P: Basta o endereço IP do hospedeiro em que o processo é executado para identificar o processo?
  - ❖ R: Não, *muitos* processos podem estar rodando no mesmo hospedeiro
- ❑ *Identificador* inclui *endereço IP* e *números de porta* associados ao processo no hospedeiro.
- ❑ Exemplos de número de porta:
  - ❖ servidor HTTP: 80
  - ❖ servidor de correio: 25

# Definições de protocolo da camada de aplicação

- ❑ tipos de mensagens trocadas,
  - ❖ p. e., requisição, resposta
- ❑ sintaxe da mensagem:
  - ❖ que campos nas mensagens & como os campos são delineados
- ❑ semântica da mensagem
  - ❖ significado da informação nos campos
- ❑ regras de quando e como processos enviam & respondem a mensagens

## protocolos de domínio público:

- ❑ definidos em RFCs
- ❑ provê interoperabilidade
- ❑ p. e., HTTP, SMTP, BitTorrent

## protocolos proprietários:

- ❑ p. e., Skype, ppstream



# Que serviço de transporte uma aplicação precisa?

## perda de dados

- ❑ algumas apls. (p. e., áudio) podem tolerar alguma perda
- ❑ outras apls. (p. e., transferência de arquivos, telnet) exigem transferência de dados 100% confiável

## temporização

- ❑ algumas apls. (p. e., telefonia na Internet, jogos interativos) exigem pouco atraso para serem "eficazes"

## vazão

- ❑ algumas apls. (p. e., multimídia) exigem um mínimo de vazão para serem "eficazes"
- ❑ outras apls. ("apls. elásticas") utilizam qualquer vazão que receberem

## segurança

- ❑ criptografia, integridade de dados,...

# Requisitos de serviço de transporte das aplicações comuns

<b>Aplicação</b>	<b>Perda de dados</b>	<b>Vazão</b>	<b>Sensível ao tempo</b>
transf. arquivos	sem perda	elástica	não
e-mail	sem perda	elástica	não
documentos Web	sem perda	elástica	não
áudio/vídeo tempo real	tolerante a perda	áudio: 5 kbps-1 Mbps vídeo: 10 kbps-5 Mbps	sim, centenas de ms
áudio/vídeo armazenado	tolerante a perda	o mesmo que antes	sim, alguns seg
jogos interativos	tolerante a perda	poucos kbps ou mais	sim, centenas de ms
Mensagem instantânea	sem perda	elástica	sim e não

# Serviços de protocolos de transporte da Internet

## serviço TCP:

- ❑ *orientado a conexão:* preparação exigida entre processos cliente e servidor
- ❑ *transporte confiável* entre processo emissor e receptor
- ❑ *controle de fluxo:* emissor não sobrecarrega receptor
- ❑ *controle de congestionamento:* regula emissor quando a rede está sobrecarregada
- ❑ *não oferece:* temporização, garantias mínimas de vazão, segurança

## serviço UDP:

- ❑ transferência de dados não confiável entre processo emissor e receptor
  - ❑ não oferece: preparação da conexão, confiabilidade, controle de fluxo, controle de congest., temporização, garantia de vazão ou segurança
- P: por que se incomodar? Por que existe um UDP?

# Aplicações da Internet: aplicação, protocolos de transporte

REDES DE  
COMPUTADORES  
E A INTERNET 5ª edição

*Uma Abordagem Top-Down*

<b>Aplicação</b>	<b>Protocolo da camada de aplicação</b>	<b>Protocolo de transporte básico</b>
e-mail	SMTP [RFC 2821]	TCP
acesso remoto	Telnet [RFC 854]	TCP
Web	HTTP [RFC 2616]	TCP
transf. arquivos	FTP [RFC 959]	TCP
multimídia com fluxo contínuo	HTTP (p. e., Youtube), RTP [RFC 1889]	TCP ou UDP
telefonía da Internet	SIP, RTP, proprietário (p. e., Skype)	normalmente UDP



# Exercícios de Fixação

REDES DE  
COMPUTADORES  
E A INTERNET 5ª edição

Uma Abordagem Top-Down

Exercícios de Fixação, página 129-130, questões 1-9

- ❑ *Kurose*: página 129-130, questões 1-9

Trabalho: Pesquisa sobre Criptografia e o SSL

- ❑ *Disponível*: no Q-Acadêmico
- ❑ *Envio do relatório*: upload via Moodle