

Aula 4

Professores:

Anna Dolejsi Santos (UFF)

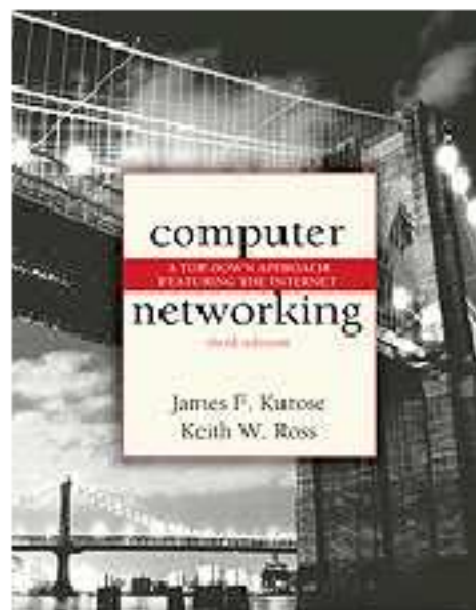
Célio Vinicius Neves de Albuquerque (UFF)

A Camada de Aplicação

Conteúdo:

- Princípios dos Protocolos da Camada de Aplicação

Livro Texto



REDES DE COMPUTADORES E A INTERNET UMA ABORDAGEM TOP-DOWN

(www.aw.com/kurose_br)

James F. Kurose e Keith W. Ross

Copyright: 2006 - 3a. Edição

ISBN: 8588639181

<http://www.pearson.com.br/>

Referências Adicionais:

Redes de Computadores

Andrew Tanenbaum

Editora Campus, 4a. Edição, 2003

ISBN: 8535211853

Obs: As figuras que não têm referências pertencem ao material disponibilizado pelo autor do livro texto ou foram produzidas pelo professor desta disciplina.

Capítulo 2: Camada de Aplicação

Metas do capítulo:

- ❑ aspectos conceituais e de implementação de protocolos de aplicação em redes
 - modelos de serviço da camada de transporte
 - paradigma cliente servidor
 - paradigma peer-to-peer
- ❑ aprenda sobre protocolos através do estudo de protocolos populares da camada de aplicação:
 - HTTP, FTP
 - SMTP/ POP3/ IMAP
 - DNS
- ❑ a programação de aplicações de rede
 - programação usando a API de sockets

Capítulo 2: Roteiro

- ❑ 2.1 Princípios dos protocolos da camada de aplicação
- ❑ 2.2 Web e HTTP
- ❑ 2.3 FTP
- ❑ 2.4 Correio Eletrônico
 - SMTP, POP3, IMAP
- ❑ 2.5 DNS
- ❑ 2.6 Compartilhamento de arquivos P2P
- ❑ 2.7 Programação de Sockets com TCP
- ❑ 2.8 Programação de Sockets com UDP

Algumas aplicações de rede

- ☐ E-mail
- ☐ Web
- ☐ Instant messaging
- ☐ Login remoto
- ☐ Compartilhamento de arquivos P2P
- ☐ Jogos de rede multi-usuários
- ☐ Vídeo-clipes armazenados
- ☐ Voz sobre IP
- ☐ Vídeo conferência em tempo real
- ☐ Computação paralela em larga escala

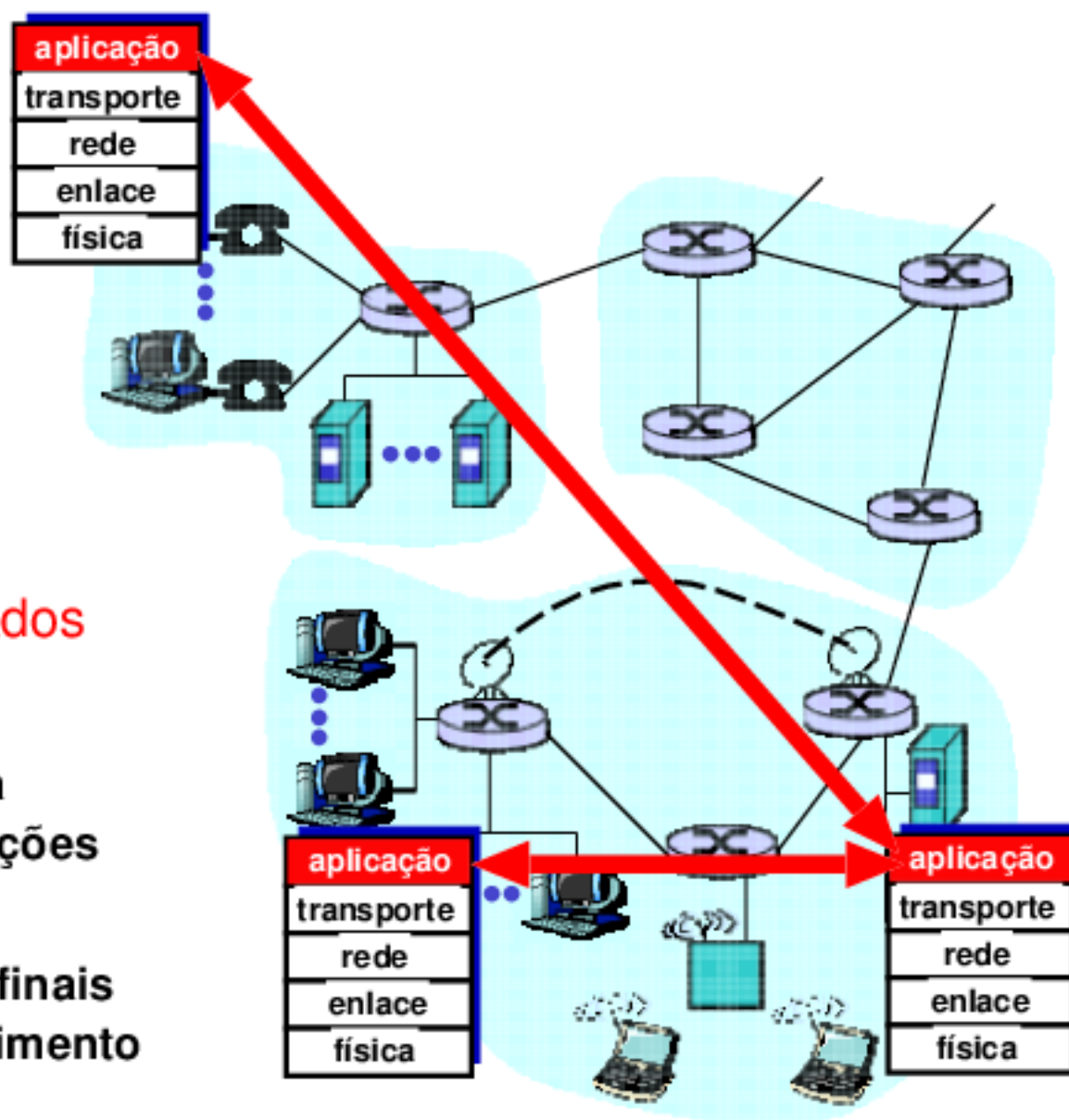
Criando uma aplicação de rede

❑ Programas que

- Executam em diferentes sistemas finais
- Comunicam-se através da rede
- p.ex., Web: servidor Web se comunica com o navegador

❑ Programas não relacionados ao núcleo da rede

- Dispositivos do núcleo da rede não executam aplicações de usuários
- Aplicações nos sistemas finais permite rápido desenvolvimento e disseminação



Arquiteturas das aplicações

- ☐ Cliente-servidor
- ☐ Peer-to-peer (P2P)
- ☐ Híbrido de cliente-servidor e P2P

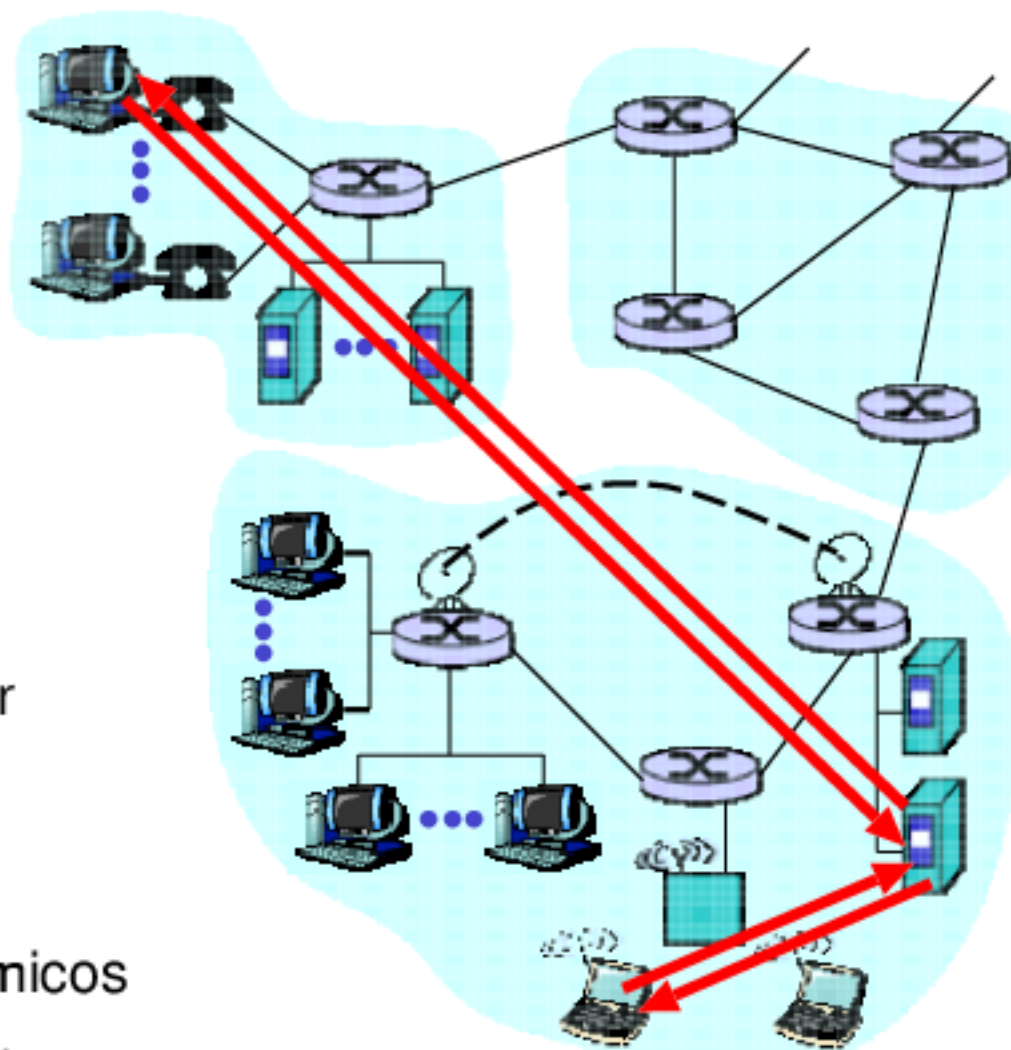
Arquitetura cliente-servidor

Servidor:

- ☐ Sempre ligado
- ☐ Endereço IP permanente
- ☐ Escalabilidade com *server farms*

Cliente:

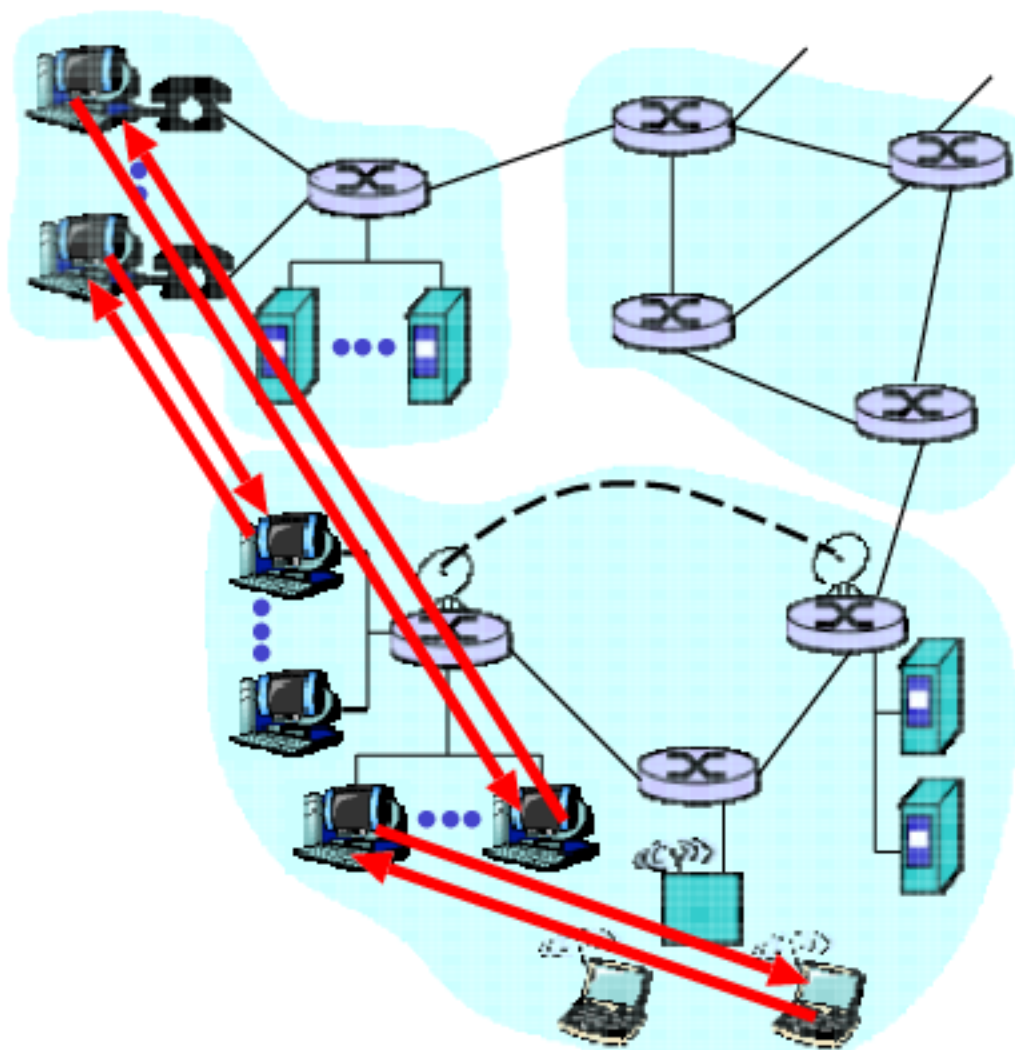
- ☐ Comunica-se com o servidor
- ☐ Pode estar conectado intermitentemente
- ☐ Pode ter endereços IP dinâmicos
- ☐ Não se comunica diretamente com outros clientes



Arquitetura P2P pura

- ❑ Não há servidor sempre ligado
- ❑ Sistemas finais arbitrários se comunicam diretamente
- ❑ Pares estão conectados intermitentemente e mudam endereços IP
- ❑ Exemplo: Gnutella

Altamente escalável
Porém, difícil de gerenciar



Híbrido de cliente-servidor e P2P

Napster

- ❑ Transferência de arquivos P2P
- ❑ Busca de arquivos centralizada:
 - Pares registram conteúdo no servidor central
 - Pares consultam o mesmo servidor central para localizar conteúdo

Instant messaging

- ❑ Conversa entre usuários P2P
- ❑ Localização e detecção de presença centralizadas:
 - Usuários registram o seu endereço IP junto ao servidor central quando ficam online
 - Usuários consultam o servidor central para encontrar endereços IP dos contatos

Processos em comunicação

Processo: programa que executa num hospedeiro

- ❑ processos no mesmo hospedeiro se comunicam usando **comunicação entre processos** definida pelo sistema operacional (SO)
- ❑ processos em hospedeiros distintos se comunicam trocando **mensagens através da rede**

Processo cliente:

processo que inicia a comunicação

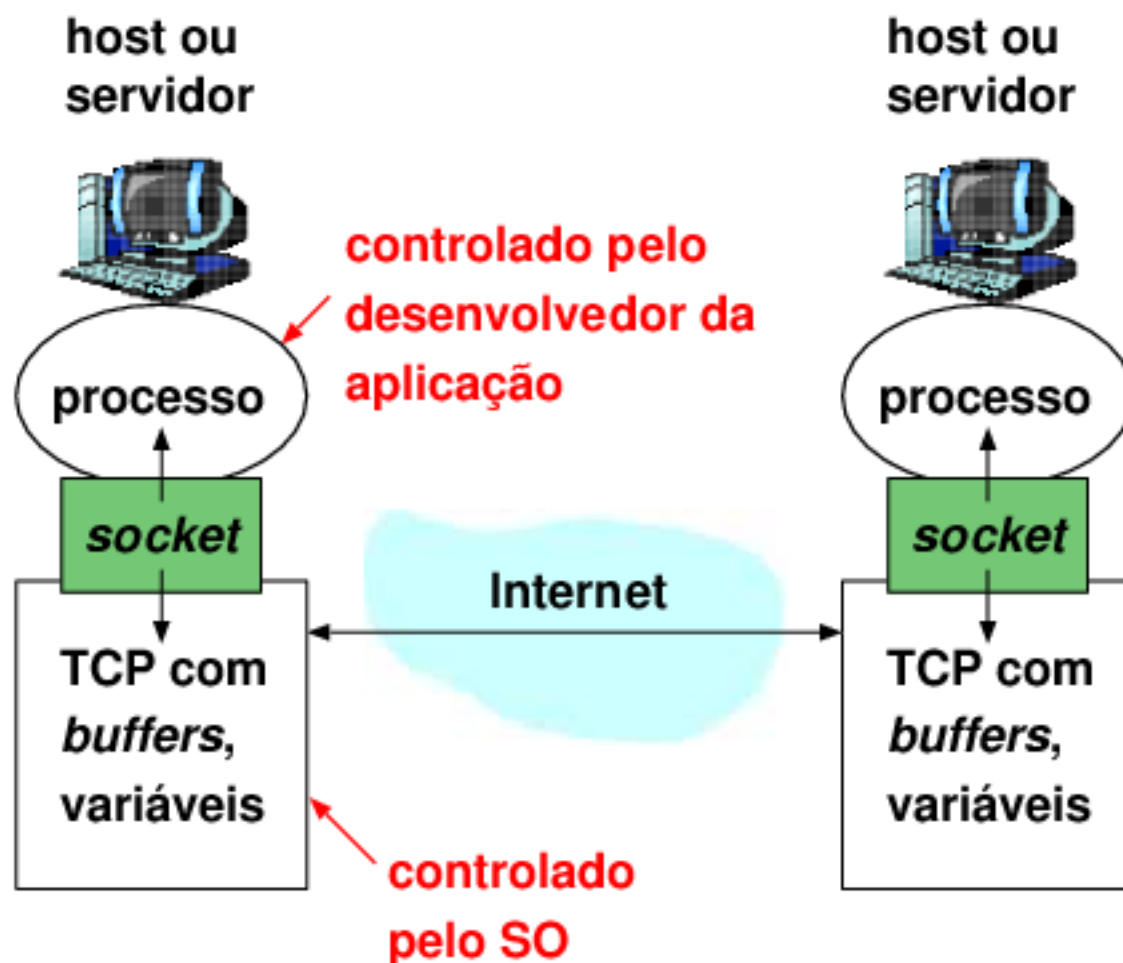
Processo servidor:

processo que espera para ser contatado

- ❑ Nota: aplicações com arquiteturas P2P possuem processos clientes e processos servidores

Sockets

- ❑ Os processos enviam/recebem mensagens para/dos seus sockets
- ❑ Um socket é análogo a uma porta
 - Processo transmissor envia a mensagem através da porta
 - O processo transmissor assume a existência da infra-estrutura de transporte no outro lado da porta que faz com que a mensagem chegue ao socket do processo receptor
- ❑ API: (1) escolha do protocolo de transporte;
(2) habilidade para fixar alguns parâmetros
(mais sobre isto posteriormente)



Endereçando os processos

- ❑ Para que um processo receba mensagens, ele deve possuir um identificador
- ❑ Cada host possui um endereço IP único de 32 bits
- ❑ **P:** o endereço IP do host no qual o processo está sendo executado é suficiente para identificar o processo?
- ❑ **Resposta:** Não, muitos processos podem estar executando no mesmo host
- ❑ O identificador inclui tanto o endereço IP quanto os **números das portas** associadas com o processo no host.
- ❑ Exemplo de números de portas:
 - **Servidor HTTP: 80**
 - **Servidor de Correio: 25**
- ❑ Mais sobre isto posteriormente.

Os protocolos da camada de aplicação definem

- ❑ Tipos de mensagens trocadas, ex. mensagens de pedido e resposta
- ❑ Sintaxe dos tipos das mensagens: campos presentes nas mensagens e como são identificados
- ❑ Semântica dos campos, i.e., significado da informação nos campos
- ❑ Regras para quando os processos enviam e respondem às mensagens

Protocolos de domínio público:

- ❑ definidos em RFCs
- ❑ Permitem a interoperação
- ❑ ex, HTTP e SMTP

Protocolos proprietários:

- ❑ Ex., KaZaA

De que serviço de transporte uma aplicação precisa?

Perda de dados

- ❑ algumas apls (p.ex. áudio) podem tolerar algumas perdas
- ❑ outras (p.ex., transf. de arquivos, telnet) requerem transferência 100% confiável

Largura de banda

- ❑ algumas apls (p.ex., multimídia) requerem quantia mínima de banda para serem "viáveis"
- ❑ outras apls ("apls elásticas") conseguem usar qq quantia de banda disponível

Temporização

- ❑ algumas apls (p.ex., telefonia Internet, jogos interativos) requerem baixo retardo para serem "viáveis"

Requisitos do serviço de transporte de apls comuns

Aplicação	Perdas	Banda	Sensibilidade temporal
transferência de arqs	sem perdas	elástica	não
correio	sem perdas	elástica	não
documentos WWW	sem perdas	elástica	não
áudio/vídeo de tempo real	tolerante	áudio: 5Kb-1Mb vídeo:10Kb-5Mb	sim, 100's mseg
áudio/vídeo gravado	tolerante	como anterior	sim, alguns segs
jogos interativos	tolerante	> alguns Kbps	sim, 100's mseg
apls financeiras	sem perdas	elástica	sim e não

Serviços providos por protocolos de transporte Internet

Serviço TCP:

- ☐ orientado a conexão: inicialização requerida entre cliente e servidor
- ☐ transporte confiável entre processos remetente e receptor
- ☐ controle de fluxo: remetente não vai "afogar" receptor
- ☐ controle de congestionamento: estrangular remetente quando a rede estiver carregada
- ☐ não provê: garantias temporais ou de banda mínima

Serviço UDP:

- ☐ transferência de dados não confiável entre processos remetente e receptor
- ☐ não provê: estabelecimento da conexão, confiabilidade, controle de fluxo, controle de congestionamento, garantias temporais ou de banda mínima

P: Qual é o interesse em ter um UDP?

Apls Internet: seus protocolos e seus protocolos de transporte

Aplicação	Protocolo da camada de apl	Protocolo de transporte usado
correio eletrônico	SMTP [RFC 2821]	TCP
acesso terminal remoto	telnet [RFC 854]	TCP
WWW	HTTP [RFC 2616]	TCP
transferência de arquivos	ftp [RFC 959]	TCP
streaming multimídia	proprietário (p.ex. RealNetworks)	TCP ou UDP
telefonia Internet	proprietário (p.ex., Dialpad)	tipicamente UDP