

Redes de Computadores

Parte 04 – camada de aplicação – introdução

Prof. Kleber Vieira Cardoso



INSTITUTO DE
INFORMÁTICA
UFG

Tópicos

- Arquiteturas de aplicações de rede
- Comunicação entre processos
- Tipos de serviços de transporte
- Serviços de transporte disponíveis na Internet
- Protocolos de aplicação
 - Aplicações de rede a serem abordadas

Algumas aplicações de rede

- *E-mail* (correio eletrônico)
- *Web*
- Mensagem instantânea
- *Login* em computador remoto (telnet ou ssh)
- Compartilhamento de arquivos P2P
- Jogos multiusuários em rede
- Vídeo em fluxo contínuo (YouTube, Hulu, Netflix)
- Voz sobre IP (Skype)
- Videoconferência em tempo real
- Redes sociais
- ...
- ...

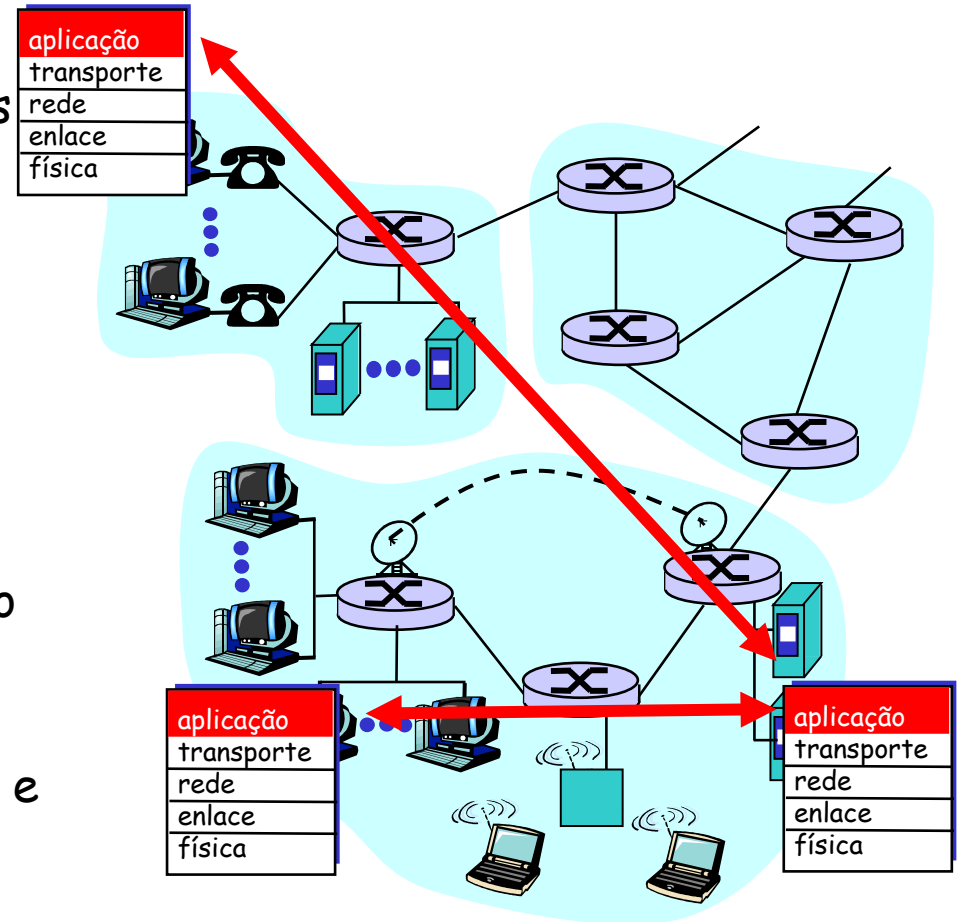
Criando uma aplicação de rede

Programas que

- Executam em diferentes sistemas finais
- Comunicam-se através da rede
- Exemplo: servidor *Web* se comunica com o navegador

Programas, em geral, não executam no núcleo da rede

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



Arquiteturas das aplicações de rede

- Cliente-servidor
- *Peer-to-peer* (P2P) ou par-a-par

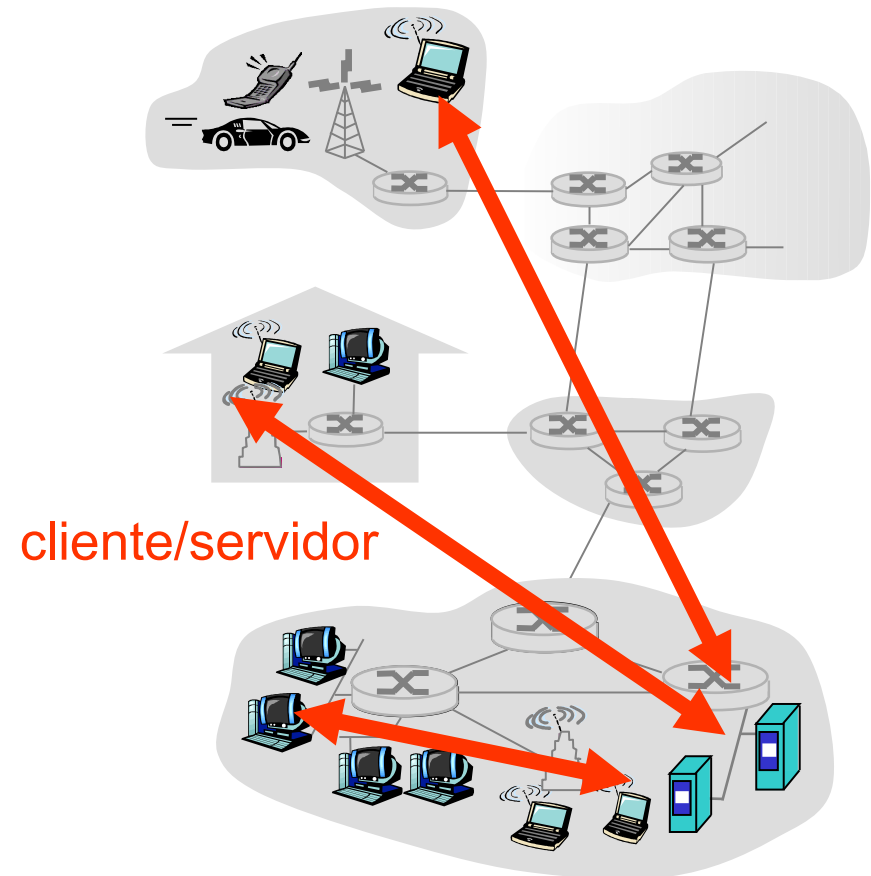
Arquitetura cliente-servidor

Servidor:

- “Sempre” ligado
- Endereço IP permanente
- Escalabilidade com centros de dados

Cliente:

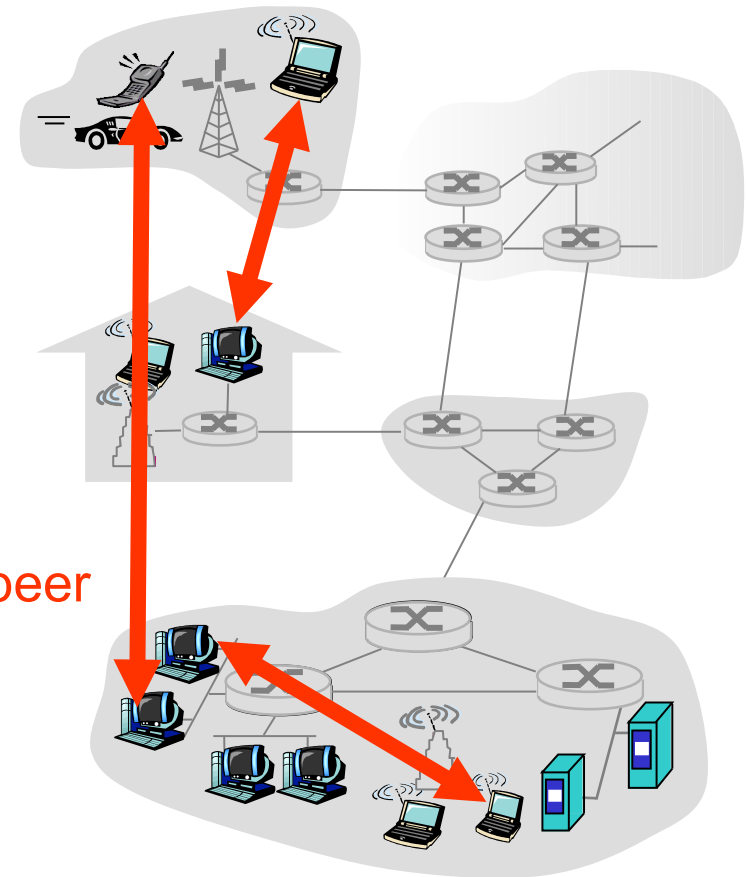
- Comunica-se com o servidor
- Pode estar conectado intermitentemente
- Pode ter endereços IP dinâmicos
- Não se comunica diretamente com outros clientes



Arquitetura P2P

- Não há servidor sempre ligado
- Sistemas finais arbitrários se comunicam diretamente
- Pares requisitam serviços de outros, fornecendo serviço em troca
- Pares estão conectados intermitentemente e mudam endereços IP

peer-to-peer



Altamente escalável

Porém, difícil de gerenciar

Comunicação entre Processos

Processo: programa que executa em um sistema final

- processos no mesmo sistema final se comunicam usando **comunicação interprocessos** definida pelo sistema operacional (SO)
 - E.g., memória compartilhada
- processos em sistemas finais distintos se comunicam trocando **mensagens** através da rede

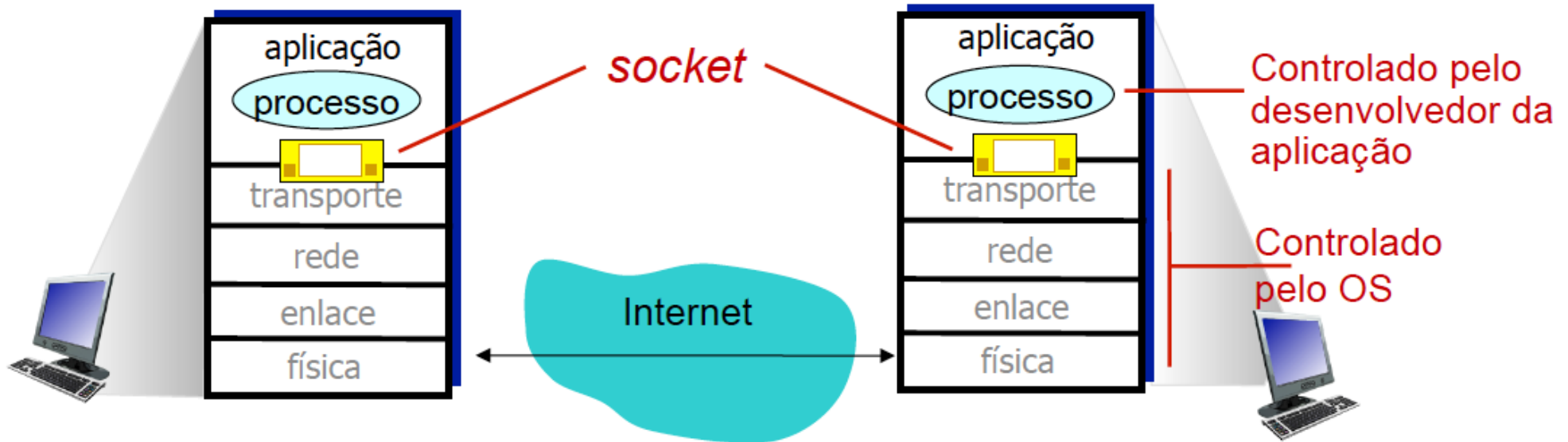
Processo cliente: processo que inicia a comunicação

Processo servidor: processo que espera ser contatado

- Nota: aplicações com arquiteturas P2P possuem processos/threads clientes e processos/threads 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 infraestrutura 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)

Endereçamento de processos

- Para que um processo receba mensagens, ele deve possuir um **identificador**
- Cada hospedeiro possui um **endereço IP** único (32 bits na versão 4 e 128 bits na versão 6)
- **P:** o endereço IP do hospedeiro no qual o processo está sendo executado é suficiente para identificar o processo?

Endereçamento de processos

- Para que um processo receba mensagens, ele deve possuir um **identificador**
- Cada hospedeiro possui um **endereço IP** único (32 bits na versão 4 e 128 bits na versão 6)
- **P:** o endereço IP do hospedeiro no qual o processo está sendo executado é suficiente para identificar o processo?
- **Resposta:** Não, muitos processos podem estar executando no mesmo hospedeiro
- O identificador inclui tanto o endereço IP quanto os **números das portas** associadas com o processo no hospedeiro
- Exemplo de números de portas:
 - Servidor HTTP: 80
 - Servidor de Correio: 25
- Para enviar uma msg HTTP para o servidor *Web* **www.inf.ufg.br** temos:
 - **Endereço IP:** 200.137.197.6
 - **Número da porta:** 80
- **Mais sobre isto posteriormente**

Os protocolos da camada de aplicação definem

- Tipos de mensagens trocadas:
 - E.g., mensagens de requisição e resposta
- Sintaxe das mensagens:
 - campos presentes nas mensagens e como são identificados
- Semântica das mensagens:
 - 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
- E.g., HTTP e SMTP

Protocolos proprietários:

- E.g., Skype, Microsoft Exchange Server

De que serviços uma aplicação necessita?

Transferência confiável de dados

- Várias aplicações (e.g., transf. de arquivos, ssh, e-mail...) requerem transferência confiável
- Algumas aplicações (e.g., áudio) podem tolerar algumas perdas

Temporização

- Algumas aplicações (e.g., VoIP, jogos interativos) requerem baixo retardo para serem "viáveis"

Largura de banda

- Algumas aplicações (e.g., vídeo de alta qualidade) requerem quantia mínima de banda para serem "viáveis"
- Outras aplicações ("aplicações elásticas") conseguem usar "qualquer" quantia de banda disponível

Segurança

- Criptografia, autenticidade, ...

Requisitos de algumas aplicações de rede

Aplicação	Perdas	Largura de Banda	Sensibilidade ao atraso
transferência de arqs	sem perdas	elástica	não
correio	sem perdas	elástica	não
documentos Web	sem perdas	elástica	não
áudio/vídeo em tempo real	tolerante	áudio: 5kb/s-1Mb/s vídeo: 10kb/s-5Mb/s	sim, 100's mseg
áudio/vídeo gravado	tolerante	como anterior	sim, alguns segs
jogos interativos	tolerante	> alguns kbps	sim, 100's mseg
mensagem de texto	sem perdas	elástica	sim (e.g., <i>chat</i>) e não (e.g., Twitter)

Serviços providos pelos protocolos de transporte da Internet

Serviço TCP:

- *orientado a conexão:* negociação 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?

Aplicações Internet: seus protocolos e seus protocolos de transporte

Aplicação	Protocolo da camada de aplicação	Protocolo de transporte usado
correio eletrônico	SMTP [RFC 2821]	TCP
acesso terminal remoto	telnet [RFC 854]	TCP
Web	HTTP [RFC 2616]	TCP
transferência de arquivos	FTP [RFC 959]	TCP
<i>streaming</i> multimídia	HTTP (e.g., YouTube) RTP [RFC 1889]	TCP ou UDP
telefonia Internet	SIP [RFC3261], RTP, proprietário (e.g., Skype)	tipicamente UDP

Aplicações de rede que serão abordadas

- *Web* (HTTP)
- Serviço de resolução de nomes (DNS)
- Aplicações P2P
 - Distribuição de arquivos
 - Vídeo de fluxo contínuo armazenado

Exercícios

- 1) Qual é a diferença entre arquitetura de rede e arquitetura de aplicação?
- 2) Para uma sessão de comunicação entre um par de processos, qual processo é o cliente e qual é o servidor?
- 3) Em uma aplicação de compartilhamento de arquivos P2P, você concorda com a afirmação: "não existe nenhuma noção de lados cliente e servidor de uma sessão de comunicação"? Justifique sua resposta.