

## T202 – B/C Redes de Computadores

### 02 – Aplicação e Transporte (Arquiteturas e Serviços)

Prof. Edson J. C. Gimenez  
*soned@inatel.br*

## Referência principal:

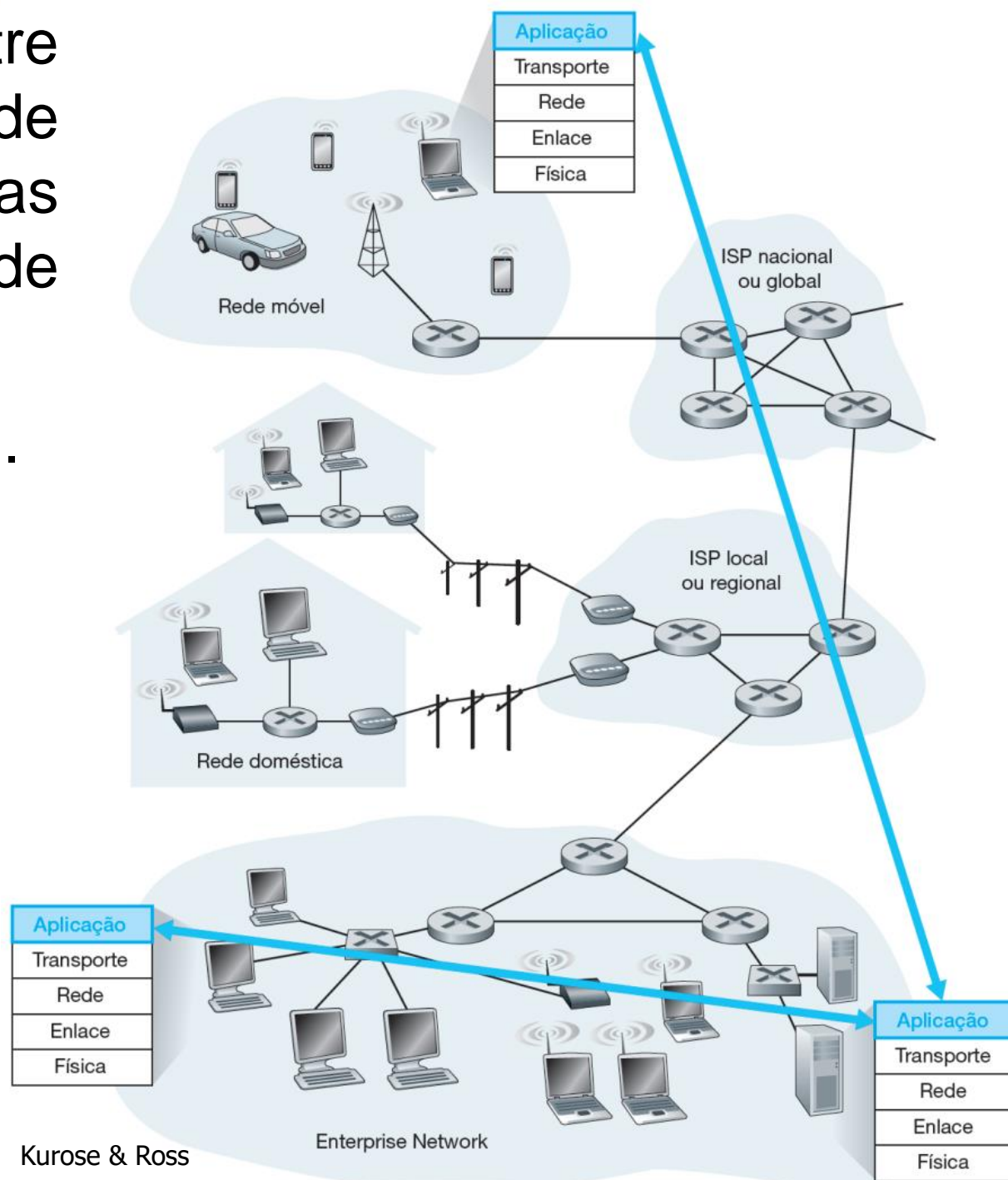
- ✓ Kurose & Ross. Redes de Computadores e a Internet: uma abordagem top-down. Capítulo 2.



## Outras referências:

- ✓ Comer. Interligação de Redes com TCP/IP; volume 1; capítulo 27.
- ✓ Tanenbaum & Wetherall. Redes de Computadores; capítulo 7.
- ✓ Farrel. A Internet e seu Protocolos: uma Análise Comparativa; capítulo 12.

- ✓ A comunicação entre duas aplicações de rede ocorre entre os sistemas finais na camada de aplicação.
- ✓ Comunicação fim a fim.

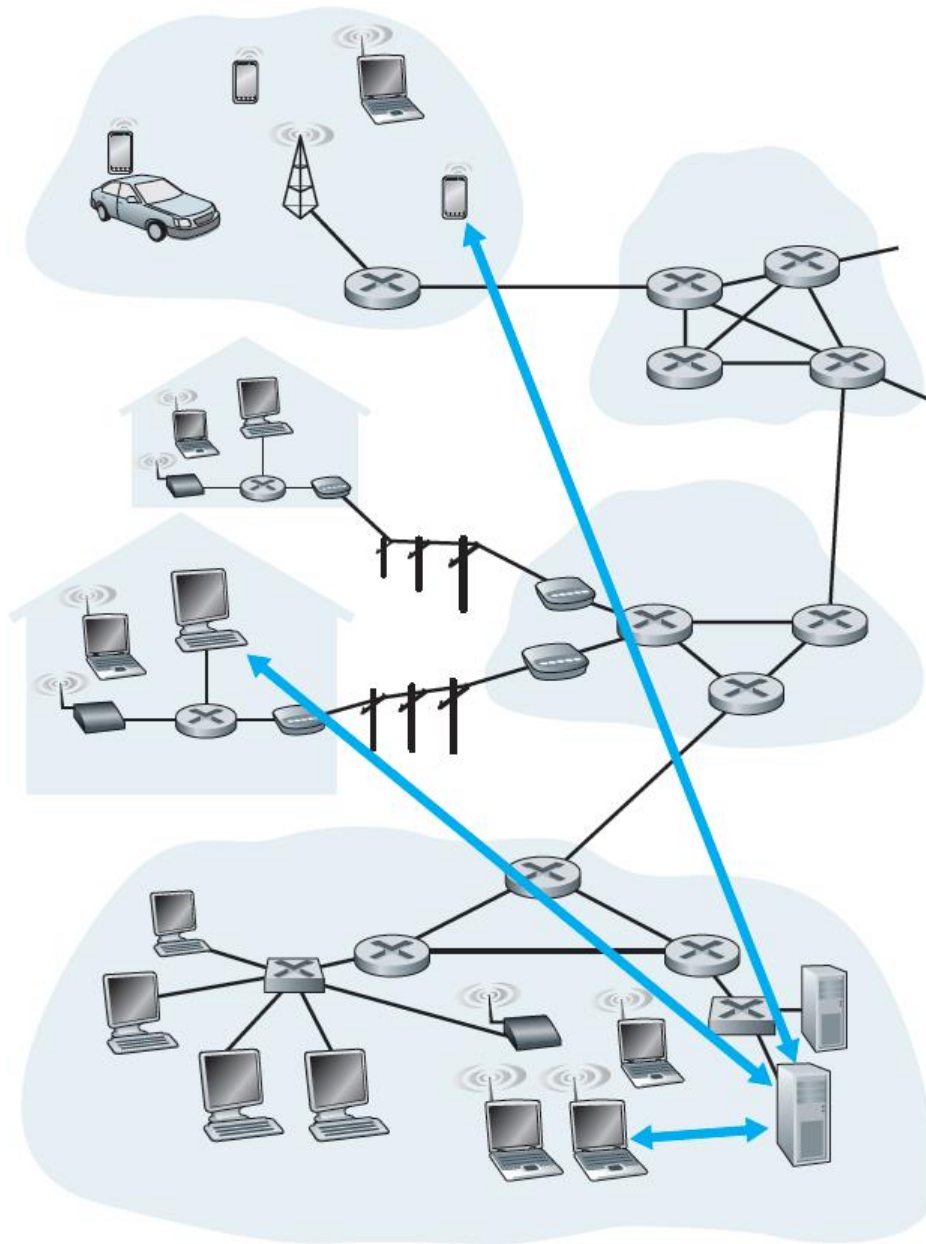


- ✓ A arquitetura de rede é fixa e provê um conjunto específico de serviços.
- ✓ A arquitetura da aplicação é projetada pelo programador e determina como a aplicação é organizada nos vários sistemas finais.
- ✓ Pode-se classificar as arquiteturas de aplicação em:
  - Arquiteturas cliente-servidor
  - Arquiteturas peer-to-peer (P2P)

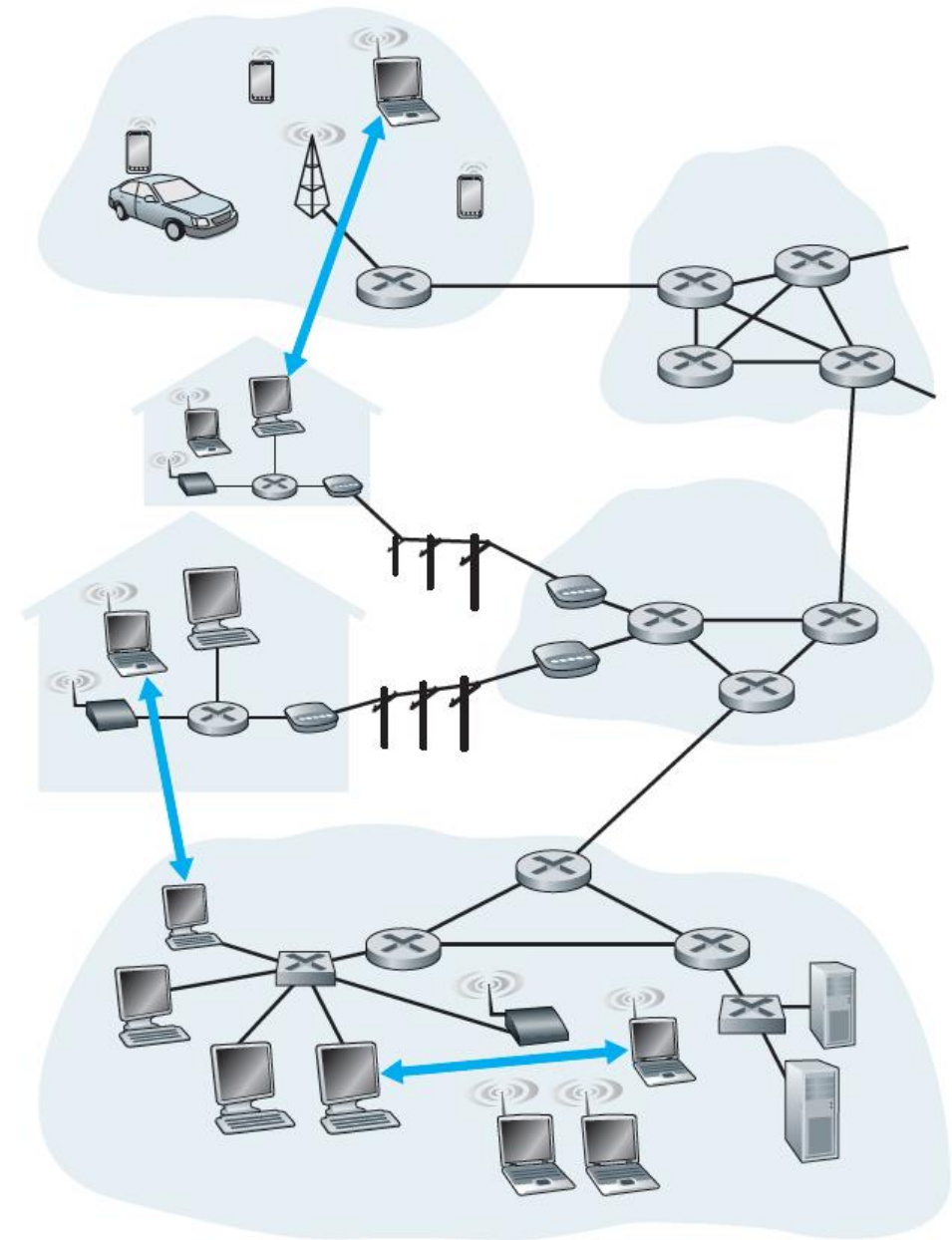
- ✓ Em uma **arquitetura cliente-servidor** há um hospedeiro sempre em funcionamento, denominado servidor, que atende a requisições de muitos outros hospedeiros, denominados clientes.
- ✓ Características:
  - Os clientes não se comunicam diretamente entre si.
  - Os servidores devem ser conhecidos; devem possuir um endereço fixo, de conhecimento público
- ✓ Exemplos: serviços Web, serviços de e-mail, etc.

- ✓ Na **arquitetura P2P** utiliza-se a comunicação direta entre duplas de hospedeiros (dispositivos finais) conectados alternadamente, denominados pares.
- ✓ Características:
  - Os pares se comunicam sem passar por nenhum servidor dedicado.
  - Tem na escalabilidade sua característica mais forte.
- ✓ Exemplos: compartilhamento de arquivos (BitTorrent), Telefonia pela Internet (Skype), TV pela Internet (IPTV), etc.





a. Arquitetura cliente-servidor

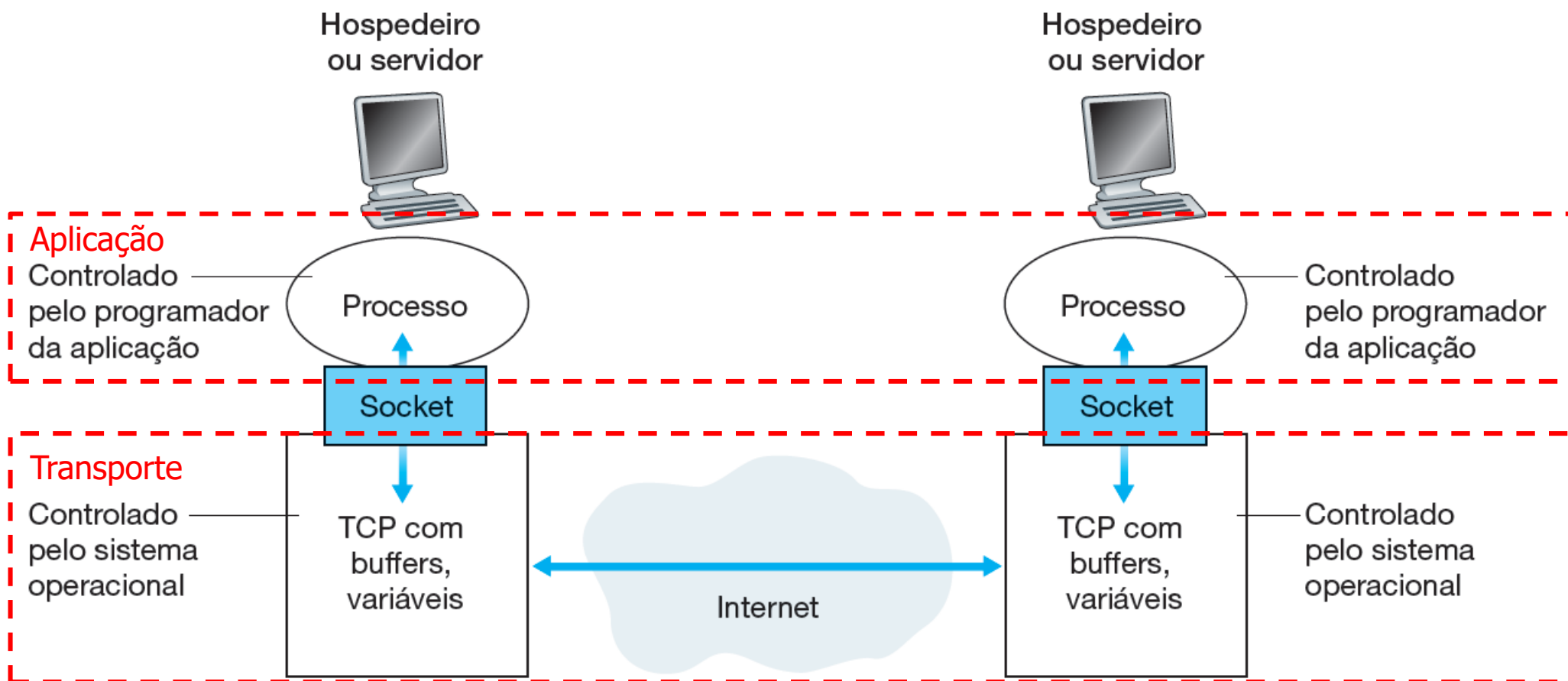


b. Arquitetura P2P

- ✓ Uma aplicação de rede consiste em pares de processos que enviam mensagens uns para os outros por meio de uma rede.
- ✓ Um processo envia mensagens para a rede e recebe mensagens da rede através de uma interface de software denominada **socket**.
- ✓ Socket é a **interface entre a camada de aplicação e a camada de transporte** em um hospedeiro.
- ✓ É também denominado **interface de programação da aplicação** (*application programming interface* — API) entre a aplicação e a rede, visto que é a interface de programação pela qual as aplicações de rede são criadas.

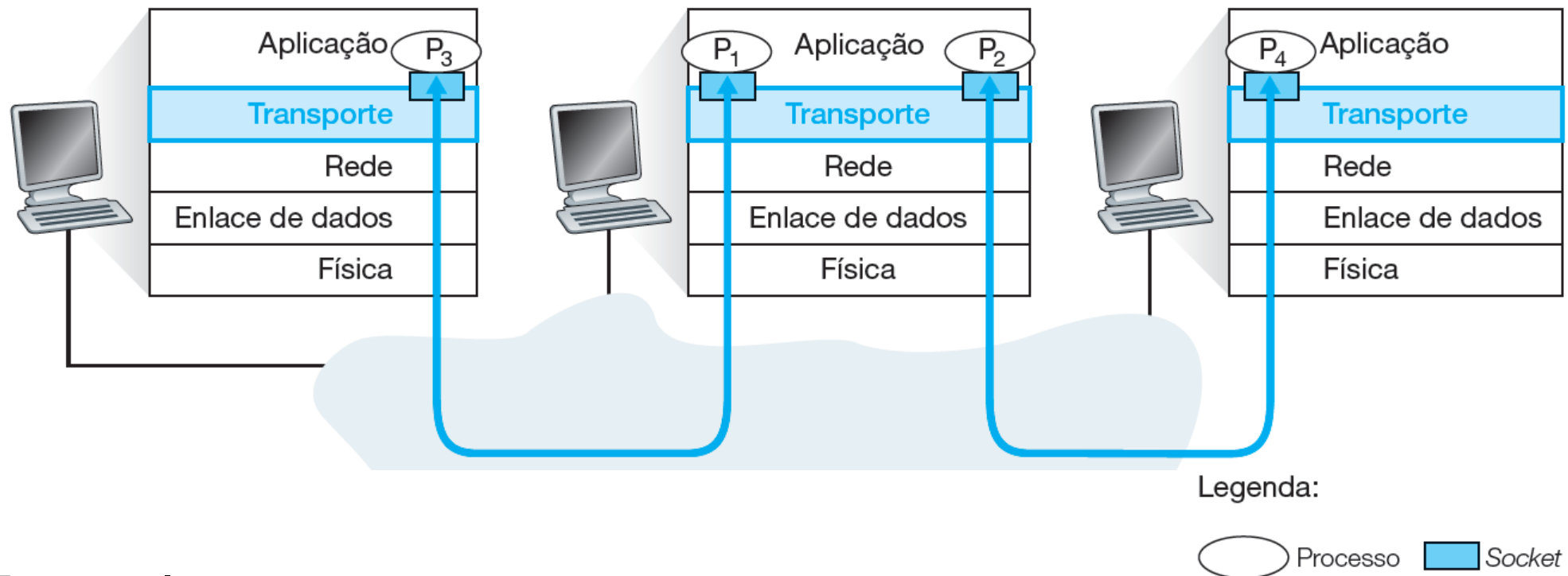


### Protocolo de aplicação, sockets e protocolo de transporte.



Adaptado: Kurose & Ross

### Protocolo de aplicação, sockets e protocolo de transporte.



### Exemplo:

- Aplicação P<sub>1</sub> e P<sub>3</sub> (uma conexão lógica) – um par de identificadores de portas.
- Aplicação P<sub>2</sub> e P<sub>4</sub> (outra conexão lógica) – um par de identificadores de portas diferentes.

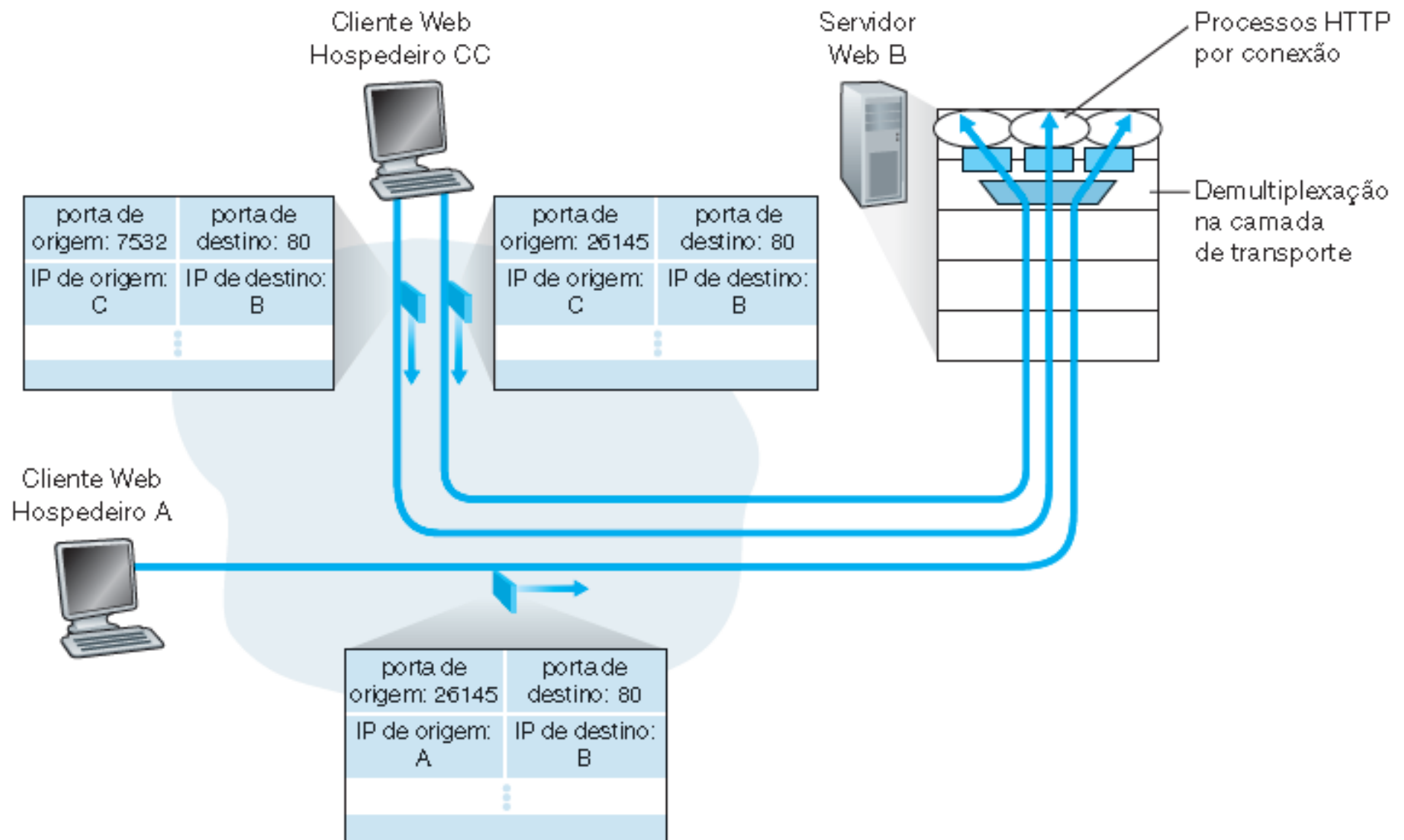
- ✓ Para identificar o processo receptor, duas informações devem ser especificadas:
  - **o endereço do hospedeiro** – um endereço IP;
  - **um identificador**, que especifica o processo receptor no hospedeiro de destino – um identificador de porta (número de porta).

- ✓ Identificadores de portas – TCP e UDP
  - Portas bem conhecidas (de 0 a 1023) – são reservados para serviços e aplicações.
  - Portas registradas (de 1024 a 49151) – são designados pela IANA para uma entidade solicitante usar com aplicações específicas.
  - Portas dinâmicas ou privadas (de 49152 a 65535) – geralmente atribuídas dinamicamente pelo sistema operacional do cliente e usadas para identificar a aplicação do cliente durante a comunicação.

- ✓ IANA - portas registradas (RFC6335)

<https://www.iana.org/assignments/service-names-port-numbers/service-names-port-numbers.xhtml>

### ✓ Identificadores de portas – TCP e UDP



### ✓ Identificadores de portas – TCP e UDP

Número da Porta	Protocolo	Aplicação	Acrônimo
20	TCP	Protocolo FTP (dados)	FTP
21	TCP	Protocolo FTP (controle)	FTP
22	TCP	Secure Shell	SSH
23	TCP	Telnet	–
25	TCP	Protocolo SMTP	SMTP
53	UDP, TCP	Serviço de Nomes de Domínio (DNS)	DNS
67	UDP	Protocolo DHCP (servidor)	DHCP
68	UDP	Protocolo DHCP (cliente)	DHCP
69	UDP	Protocolo TFTP	TFTP
80	TCP	Protocolo HTTP	HTTP
110	TCP	Protocolo POP3	POP3
143	TCP	Protocolo IMAP	IMAP
161	UDP	Protocolo SNMP	SNMP
443	TCP	Protocolo HTTPS	HTTPS (Protocolo de Transferência de Hipertexto Seguro)



### Transferência confiável de dados

- ✓ Deve garantir que os dados enviados por um processo origem sejam transmitidos correta e completamente para o processo destino.
- ✓ Quando um protocolo de transporte oferece esse serviço, o processo remetente passa seus dados para um socket e confia que eles serão entregues corretamente ao processo destinatário.
- ✓ Pode não ser necessário para aplicações tolerantes à perda, tais como aplicações multimídia.

### Vazão

- ✓ É a taxa pela qual o processo remetente pode enviar bits ao processo destinatário, podendo oscilar com o tempo.
- ✓ Aplicações sensíveis à largura de banda, tais como algumas aplicações multimídia, necessitam de uma vazão mínima garantida.
- ✓ Aplicações elásticas, tais como correio eletrônico, transferência de arquivos e transferências Web, podem fazer uso de qualquer quantidade (mínima ou máxima) disponível.

### Temporização

- ✓ Aplicações **interativas em tempo real**, como a telefonia por Internet, ambientes virtuais, teleconferência e jogos multijogadores, exigem restrições de temporização no envio de dados para garantir eficácia.
- ✓ Para aplicações que **não são em tempo real**, é sempre preferível um atraso menor a um maior, mas não há nenhuma limitação estrita aos atrasos fim a fim.

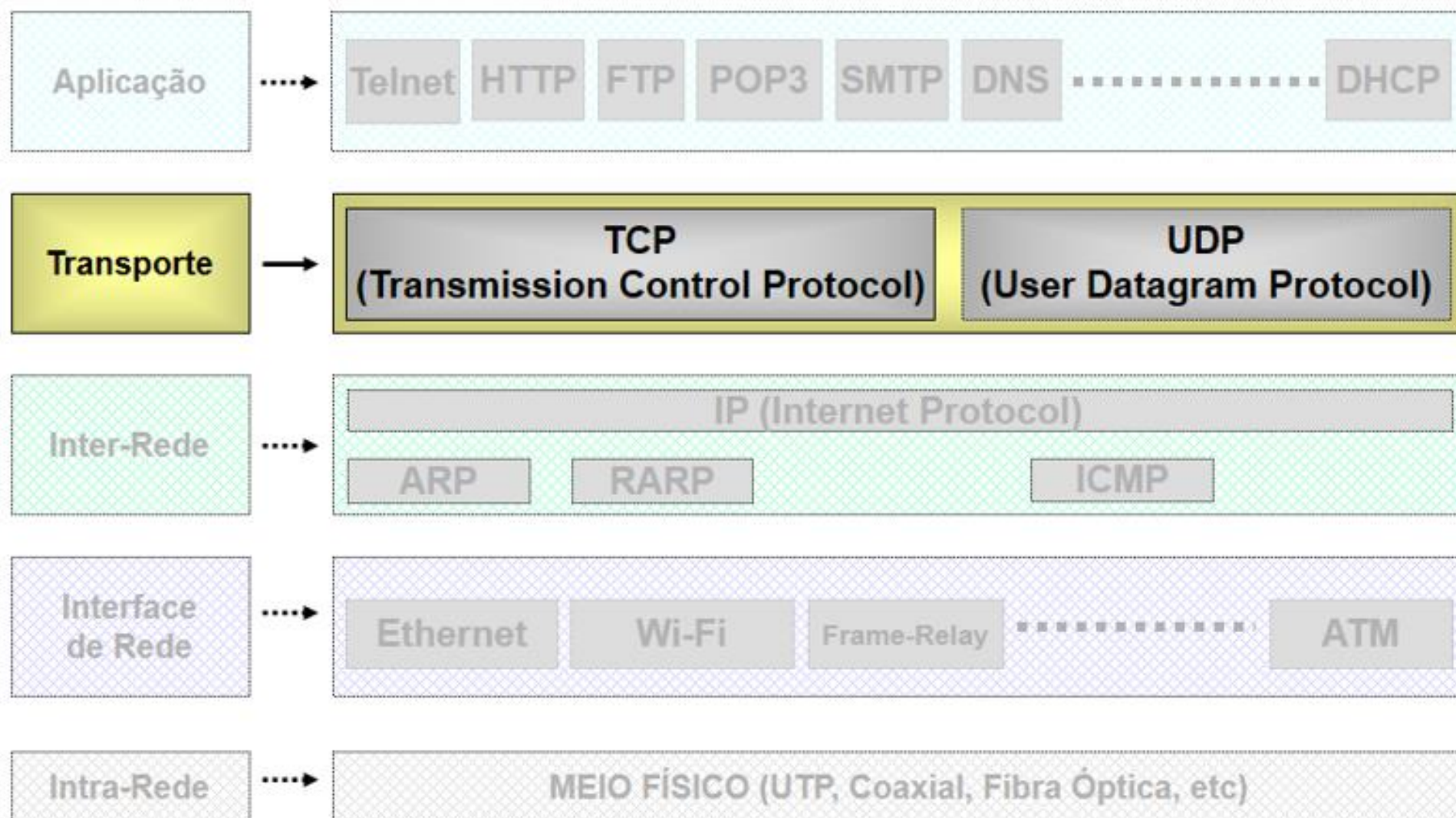
### Segurança

- ✓ Um protocolo de transporte pode, além do sigilo, fornecer outros serviços de segurança aos dados trocados entre os processos origem e destino, incluindo integridade dos dados e autenticação do ponto terminal.

### Requisitos de algumas aplicações de rede

Aplicação	Perda de dados	Vazão	Sensibilidade ao tempo
Transferência / download de arquivo	Sem perda	Elástica	Não
E-mail	Sem perda	Elástica	Não
Documentos Web	Sem perda	Elástica (alguns kbits/s)	Não
Telefonia via Internet/ videoconferência	Tolerante à perda	Áudio: alguns kbits/s – 1Mbit/s Vídeo: 10 kbits/s – 5 Mbits/s	Sim: décimos de segundo
Áudio/vídeo armazenado	Tolerante à perda	Igual acima	Sim: alguns segundos
Jogos interativos	Tolerante à perda	Poucos kbits/s – 10 kbits/s	Sim: décimos de segundo
Mensagem instantânea	Sem perda	Elástico	Sim e não

- ✓ A Internet (pilha TCP/IP) disponibiliza dois protocolos de transporte para as aplicações: o TCP (*Transmission Control Protocol*) e o UDP (*User Data Protocol*).





## Serviços do TCP

- ✓ O modelo de serviço TCP inclui um **serviço orientado para conexão** e um **serviço confiável** de transferência de dados.
  - **Serviço orientado para conexão:** o TCP faz com que o cliente e o servidor troquem informações de controle de camada de transporte antes que as mensagens de camada de aplicação comecem a fluir.
  - **Serviço confiável de transporte:** os processos comunicantes contam com o TCP para a entrega de todos os dados enviados sem erro e na ordem correta ao processo destino
- ✓ O TCP inclui ainda um mecanismo de controle de congestionamento, que limita a capacidade de transmissão de um processo (cliente ou servidor) quando a rede esta congestionada entre remetente e destinatário.

## Serviços do UDP

- ✓ O UDP é um protocolo de transporte **não orientado para conexão**
  - Nenhuma troca de informações ocorre antes que os dois processos comecem a se comunicar (troca de dados).
  - É um protocolo simplificado, leve, com um modelo de serviço minimalista.
- ✓ O UDP provê um **serviço não confiável** de transferência de dados
  - Quando um processo envia uma mensagem para um *socket* UDP, o protocolo não oferece garantias de que a mensagem chegará ao processo receptor.
  - Além disso, mensagens que chegam de fato ao processo receptor podem chegar fora de ordem.
- ✓ O UDP não inclui nenhum mecanismo de controle de congestionamento.

## Serviços não oferecidos:

- ✓ Tanto o TCP quanto o UDP **não oferecem garantias de vazão e de temporização**, que podem ser necessário para algumas aplicações.
- ✓ Neste caso, as aplicações devem, já em seu desenvolvimento, serem projetadas para lidar com suas necessidades.

## Atividade 3 – Aplicação/Transporte (Arquiteturas/Serviços)

Responda de forma clara e objetiva:

- 1) Considerando o tráfego de mensagens entre dois dispositivos na Internet, temos diferentes tipos de atraso. Cite três desses possíveis atrasos, diferenciando-os.
- 2) Qual a diferença básica entre as arquiteturas “cliente-servidor” e “peer-to-peer”.
- 3) De forma simples, como pode ser definido um socket?
- 4) Qual a diferença básica entre os serviços orientados e não orientados à conexão?
- 5) Qual a diferença básica entre os serviços “confiável” e “não confiável”?
- 6) Qual a razão para aplicações em tempo real, em sua maioria, fazerem uso do protocolo UDP na camada de Transporte?