

Redes de Computadores

Camada de Transporte

Objetivo

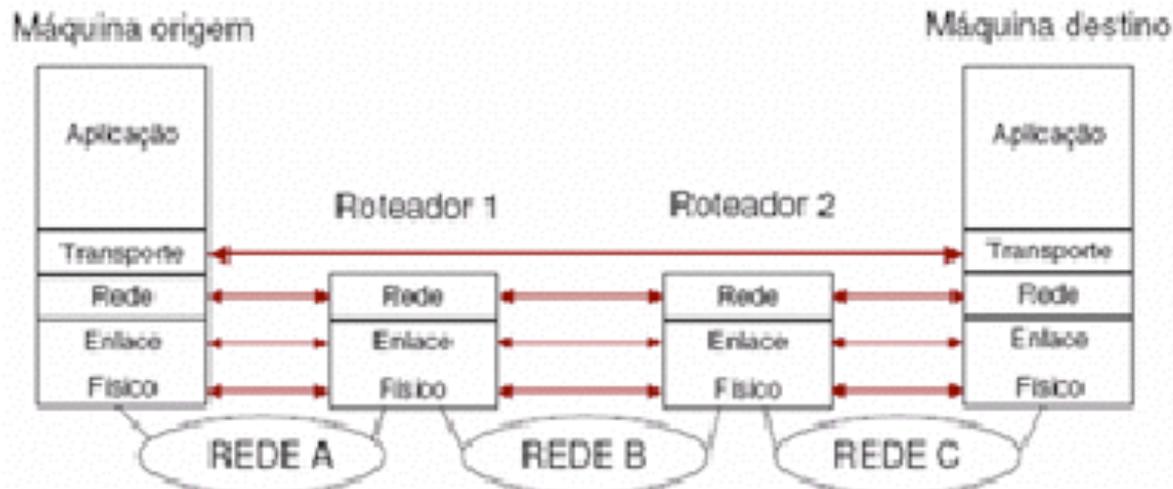
- Apresentar as características da camada de transporte da arquitetura TCP/IP
- Apresentar os serviços fornecidos pela camada de transporte
- Estudar os protocolos da camada de transporte
 - Serviço de datagramas
 - Serviço de circuito virtual

Camada de transporte

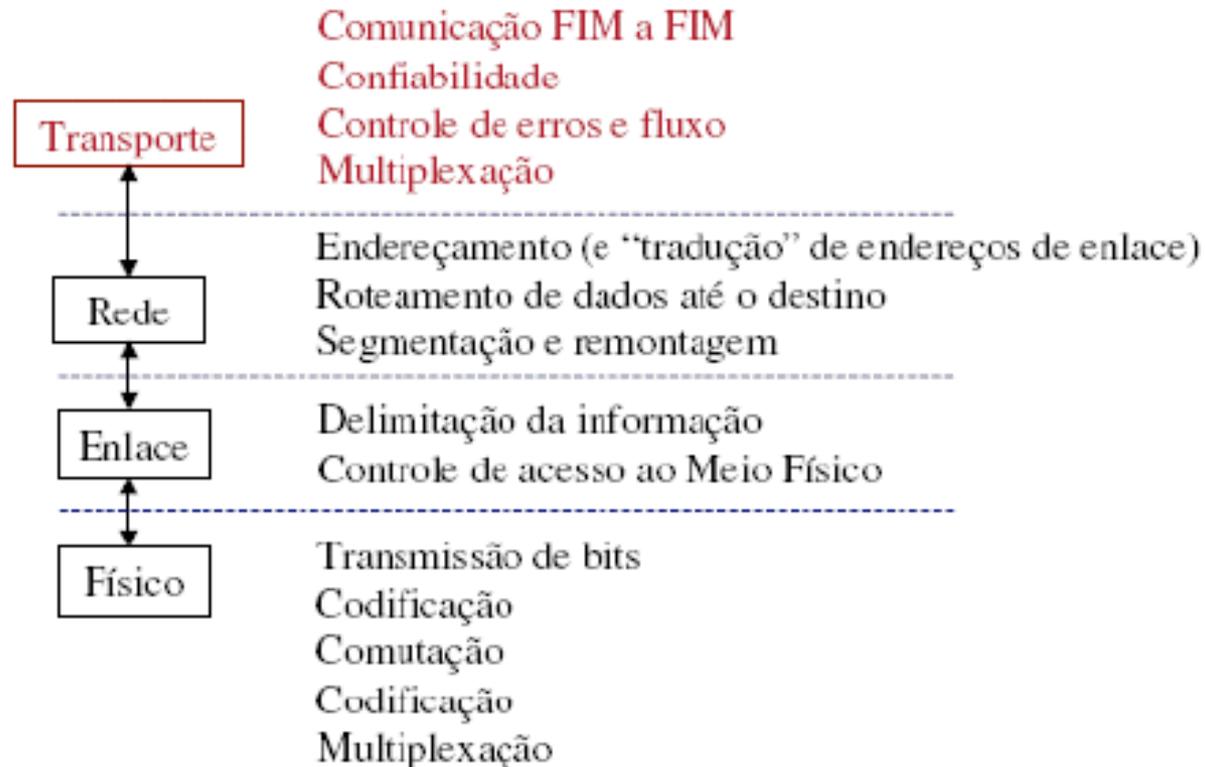
- Objetivo
 - Prover a comunicação fim-a-fim entre os processos de aplicação
- Funcionalidades
 - Serviço de datagrama
 - Serviço de circuito virtual
 - Identificação de processos

Camada de transporte

- A entidade de transporte da máquina de origem se comunica diretamente com a entidade de transporte da máquina de destino, independente de todos os sistemas intermediários entre eles
 - Nos níveis físico, enlace e rede isto não é possível



Camada de transporte



Serviço de datagrama

■ Característica

■ Serviço não confiável

- Não garante a entrega dos datagramas
- Pode perder e retardar datagramas
- Provê apenas a detecção de erros, garantindo a integridade dos dados

■ Serviço sem conexão

- Datagramas são individuais e independentes
- Seqüência dos datagramas não é assegurada

Serviço de circuito virtual

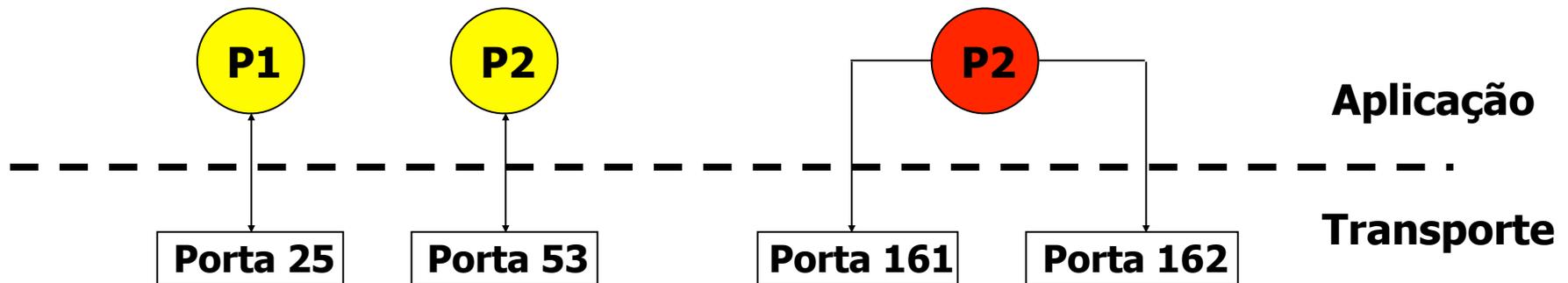
■ Característica

- Serviço orientado a fluxo
 - Divide o fluxo de dados em seguimentos
- Serviço confiável
 - Garante a entrega do fluxo de dados na seqüência correta e sem erros
 - Prevê controle de erro, seqüência e fluxo
- Serviço orientado à conexão
 - Negocia parâmetros operacionais na abertura da conexão
 - Conexões full-duplex

Identificação de processos

■ Portas

- Número inteiro positivo que representa um ponto de comunicação (0 a 65535)
- Processos são associados à portas
- Par (endereço IP, Porta) identifica unicamente cada ponto de comunicação



Composição

■ Protocolos

■ UDP (User Datagram Protocol)

- Provê o serviço de datagramas não confiável e sem conexão
- Pode ser visto como uma extensão do protocolo IP que entrega datagramas entre processos

■ TCP (Transfer Control Protocol)

- Provê o serviço de circuito virtual confiável e orientado à conexão

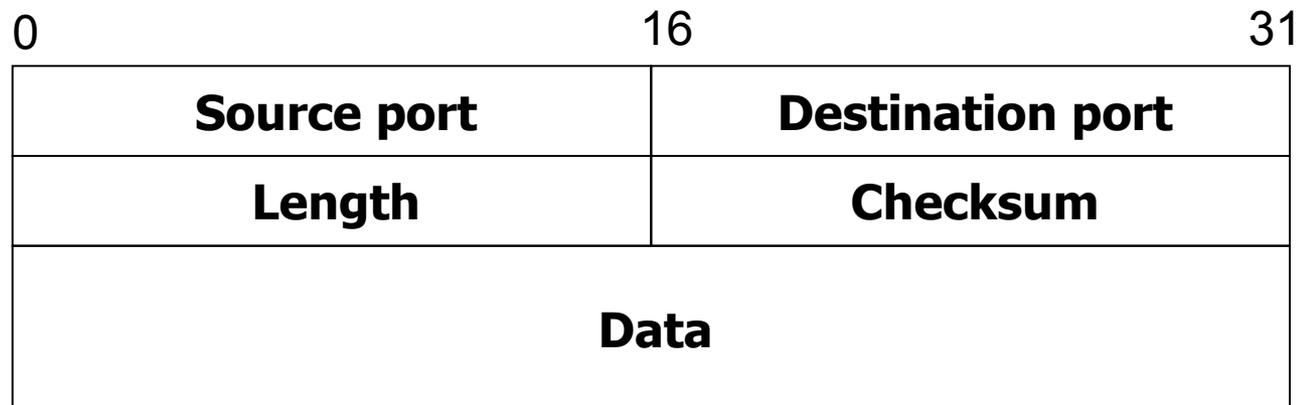
Protocolo UDP

■ Fundamentos

- Define a unidade de dados do serviço de datagramas, denominada datagrama UDP
 - Especifica o formato e a função dos campos
- Multiplexa mensagens geradas nos processos na camada de rede
 - Encapsula datagramas UDP em datagramas IP
- Demultiplexa datagramas UDP para os respectivos processos destino
 - Extrai mensagens dos datagramas UDP

Protocolo UDP

- Formato do datagrama
 - Cada datagrama é tratado de forma individual e independente
 - Pode ser enviados por diferentes rotas



Protocolo UDP

■ Campos do datagrama

■ Length

- Tamanho total do datagrama em bytes (cabeçalho + dados)

■ Checksum

- Assegura a integridade do datagrama
- Inclui o cabeçalho e os dados
- Detecção de erros é opcional

■ Data

- Dados do datagrama

Source port	Destination port
Length	Checksum
Data	

Protocolo TCP

■ Fundamentos

- Define a unidade de dados do serviço de circuito virtual, denominada seguimento TCP
 - Especifica o formato e a função dos campos
- Multiplexa mensagens geradas pelos processos no serviço da camada de rede
 - Encapsula segmentos em datagramas IP
- Demultiplexa segmentos para os respectivos processos destino
 - Extrai mensagens dos segmentos

Protocolo TCP

■ Fundamentos

- Adota uma abordagem baseada em fluxo de dados (data stream)
 - Trata o fluxo de dados como uma cadeia contínua de bytes
 - Decide como agrupar bytes em segmentos
- Adota uma abordagem orientada à conexão full-duplex
 - Estabelecimento da conexão
 - Transferência de dados
 - Fechamento da conexão

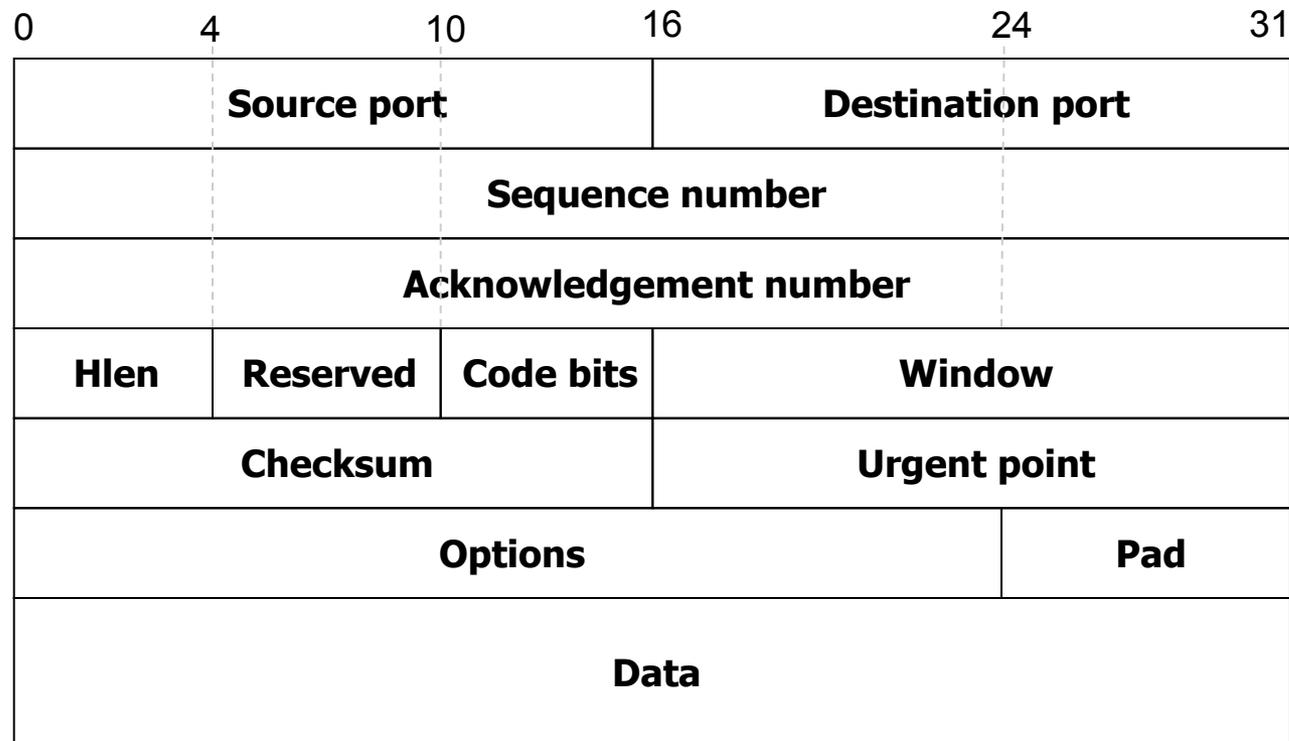
Protocolo TCP

■ Fundamentos

- Define mecanismos integrados de controle de erro e seqüência
 - Asseguram a entrega do fluxo de dados na seqüência correta e sem erros
- Define mecanismo de controle de fluxo
 - Regula e compatibiliza a taxa de transmissão das unidades envolvidas
 - Evita descarte de segmentos por falta de recursos da estação destino

Protocolo TCP

■ Formato do segmento TCP



Protocolo TCP

■ Campos do segmento

■ Hlen

- Tamanho do cabeçalho em unidades de 4 bytes

■ Reserved

- Reservado para uso futuro (Não utilizado)

■ Checksum

- Assegura a integridade do segmento
- Considera um pseudocabeçalho e pode incluir um pad (0)

Protocolo TCP

■ Campos do segmento

■ Code bits

■ Indica propósito e conteúdo do segmento

- URG: Dados urgentes
- ACK: reconhecimento
- PSH: mecanismo de push
- RST: abordo de conexão (reset)
- SYN: Abertura de conexão
- FIN: fechamento de conexão



Protocolo TCP

- Campos do segmento
 - Options
 - Lista variável de informações opcionais
 - MSS – Maximum Segment Size
 - Torna o tamanho do cabeçalho variável
 - Padding
 - Bits 0 que tornam o segmento múltiplo de 32 bits
 - Data
 - Dados do segmento

Protocolo TCP

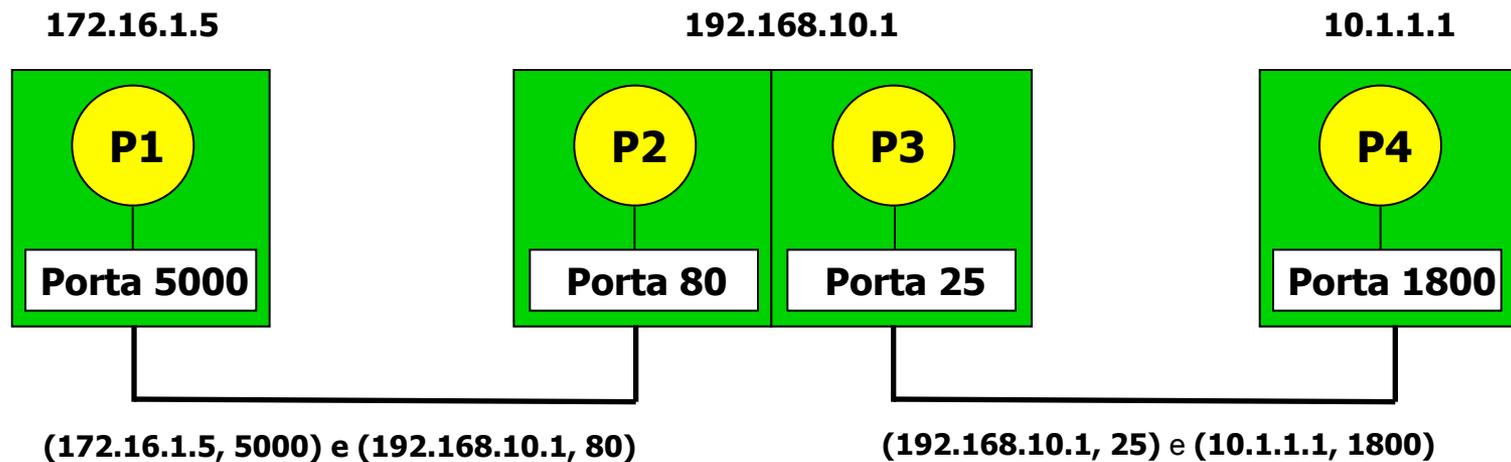
■ Portas

- Source port
 - Porta associada ao processo de origem
- Destination port
 - Porta associada ao processo de destino
- Endpoint
 - Definido pelo par (Endereço IP, porta)
 - Identifica de forma única cada porta ou ponto de comunicação na inter-rede

Protocolo TCP

■ Conexão

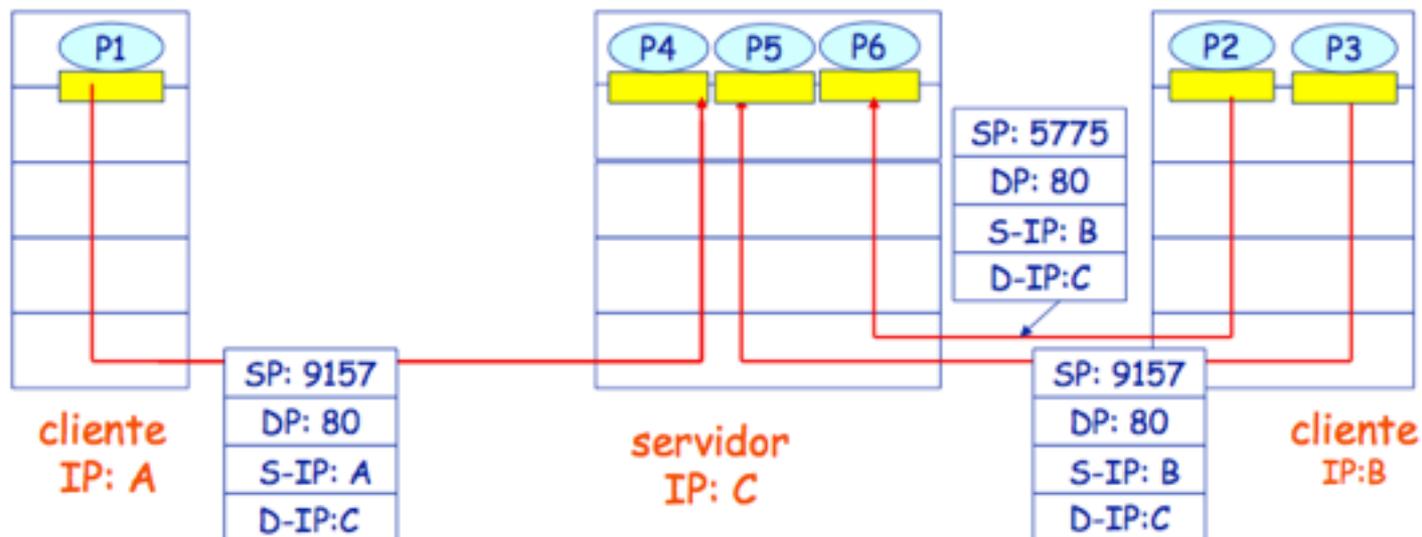
- Cada conexão é identificada por um par de endpoints
- Também conhecida como Socket pair
- Várias conexões por estação



Protocolo TCP

■ Conexão

- Cada endpoint local **pode** participar de diversas conexões com endpoints remotos
 - Compartilhamento de endpoints
 - O Sistema Operacional deve garantir que o par de endpoint da conexão é único

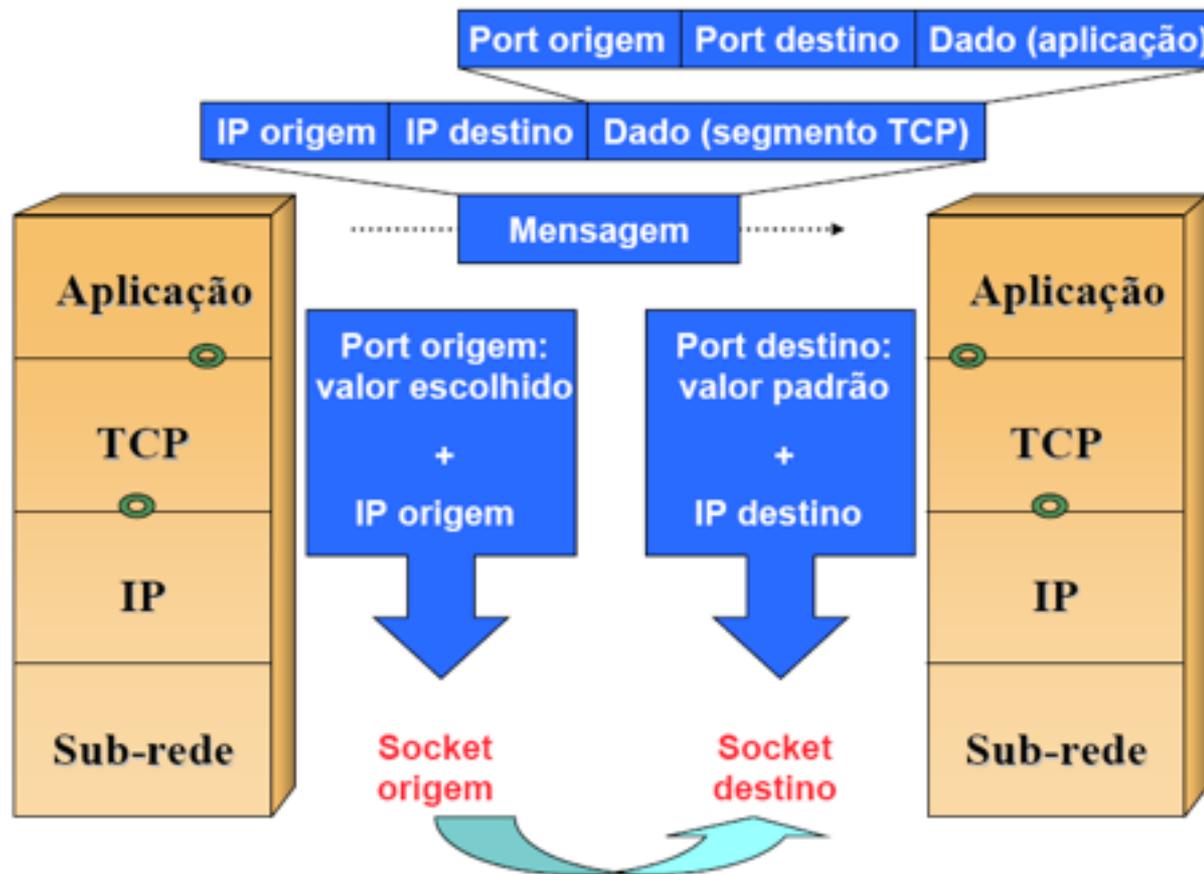


Protocolo TCP

- Demultiplexação de mensagens
 - Segmentos recebidos são associados às **conexões**, não apenas as portas
 - Avalia o par de endpoints da conexão
 - Portas origem e destino são obtidas do segmento recebido
 - Endereço IP origem e destino são obtidos do datagrama IP
 - Cada conexão possui um buffer de transmissão e um Buffer de recepção em cada extremidade

Protocolo TCP

- Processo de estabelecimento de conexões

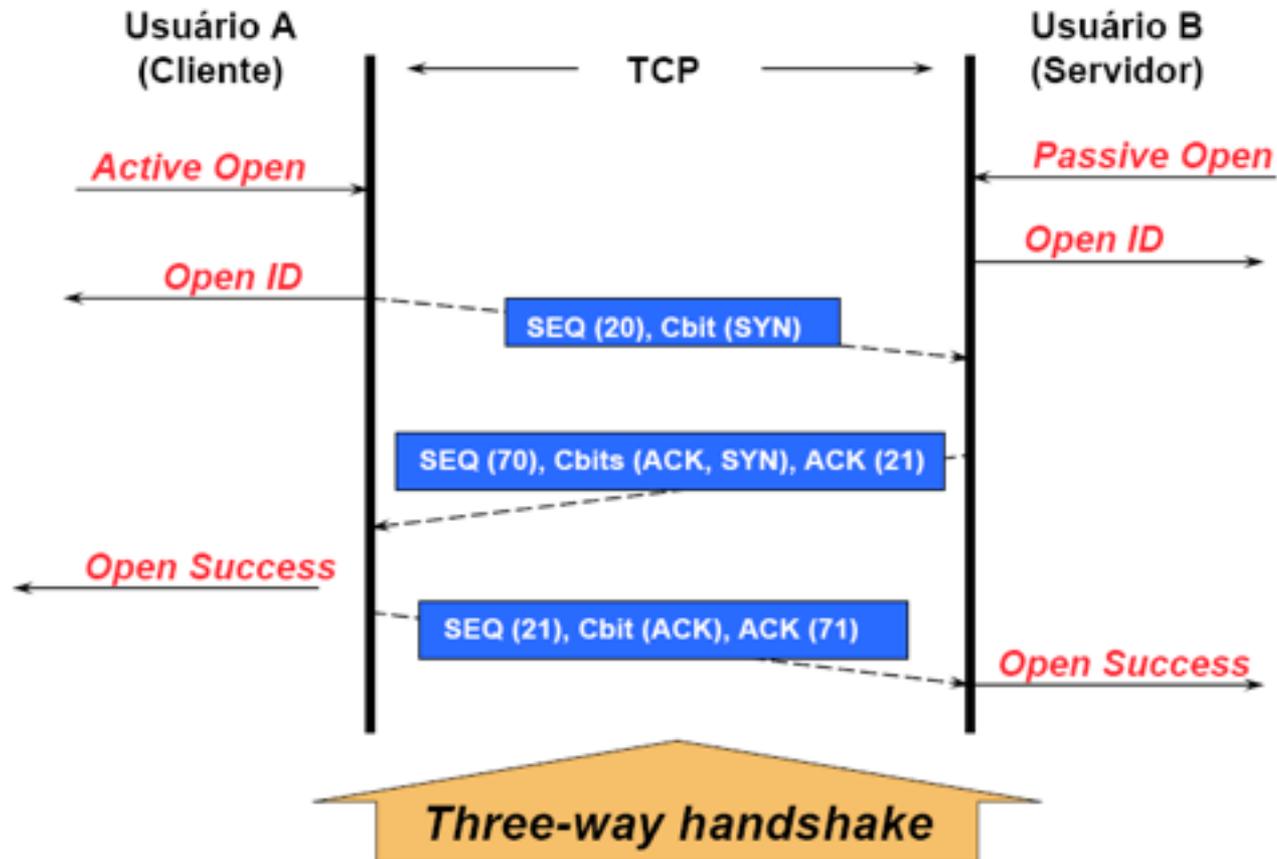


Protocolo TCP

- Estabelecimento de conexões
 - Three way handshake
 - Negocia e sincroniza o valor inicial dos números de seqüência em ambas as direções
 - Baseado na arquitetura cliente-servidor
 - O servidor deve está com a porta aberta em estado de escuta (Listening)

Protocolo TCP

■ Estabelecimento de conexões

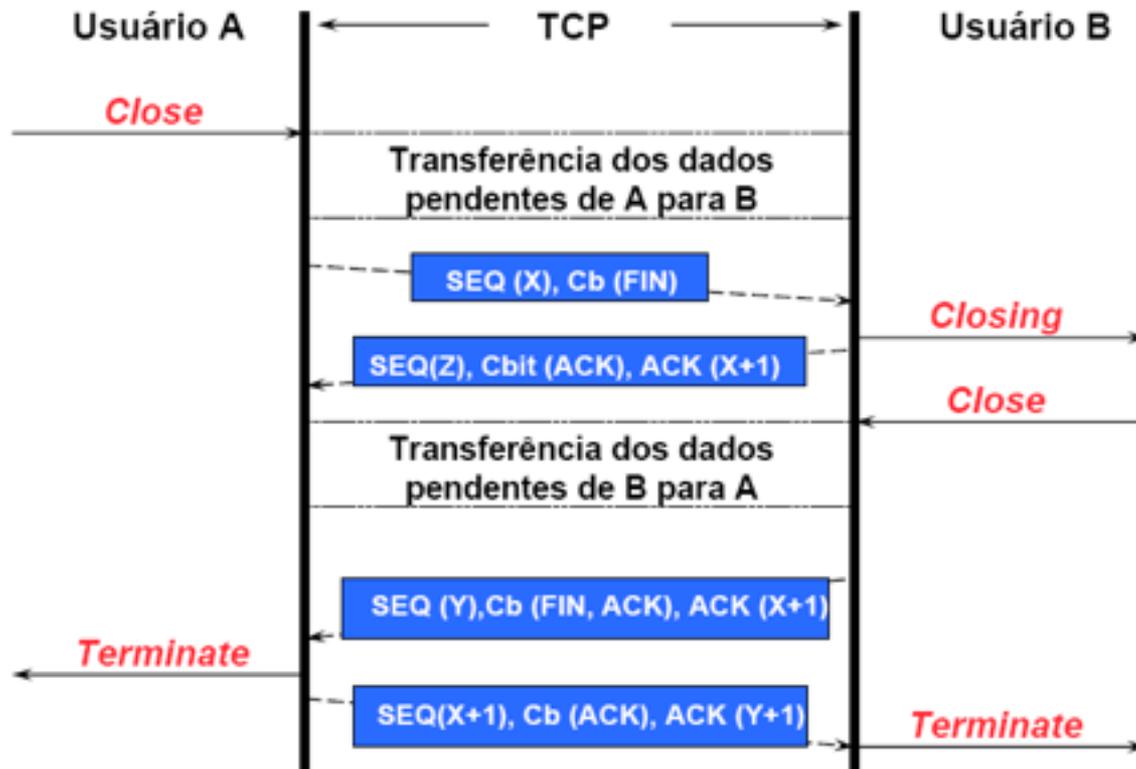


Protocolo TCP

- Transmissão de dados
 - Entrega de dados “fora-de-banda”
 - Campo Urgent Point
 - o transmissor transmite o dado urgente na área de dados e seta o bit URG (campo Codebits), indicando a posição no segmento onde o dado urgente terminou
 - O receptor deve notificar a aplicação sobre a chegada do dado urgente tão logo quanto possível
 - Mecanismo de Push
 - Aplicação avisa ao TCP para enviar o dado imediatamente
 - Força a geração de um segmento com os dados já presentes no Buffer
 - Não aguarda o preenchimento do Buffer
 - Segmentos gerados pelo mecanismo de PUSH são marcados com o flag PSH no campo codebits

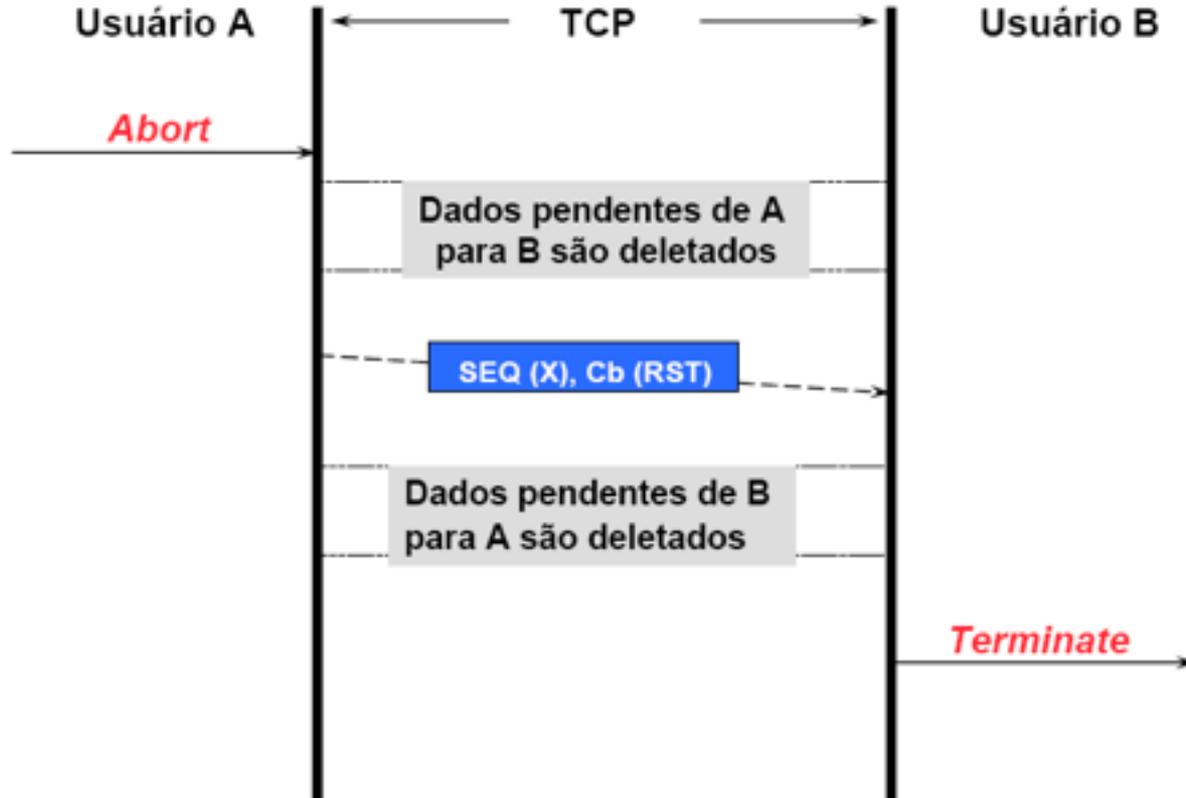
Protocolo TCP

- Fechamento de conexão (Liberação ordenada)
 - Ocorre separadamente em cada direção da conexão



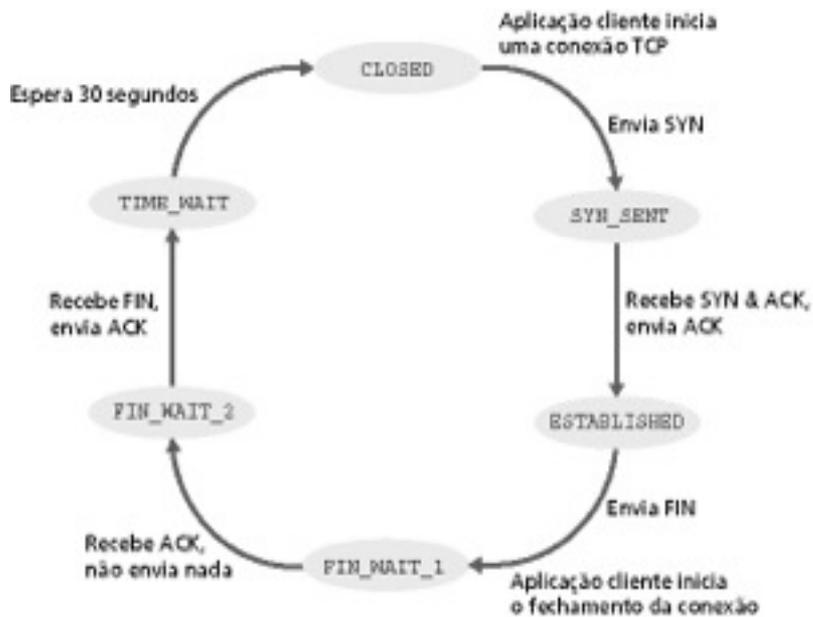
Protocolo TCP

- Fechamento de conexão (Término abrupto)

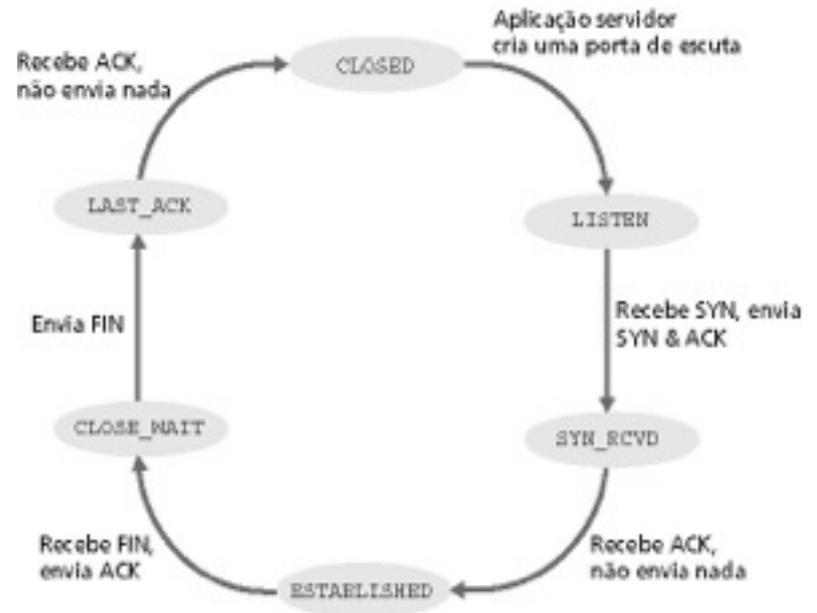


Protocolo TCP

- Estados das conexões



Estados do cliente



Estados do servidor

Protocolo TCP

■ Controle de seqüência

- Fluxo de dados é tratado como uma seqüência de bytes
 - Cada byte possui um número de seqüência
 - Numeração nem sempre começa em 0 (zero)
 - Negociado no estabelecimento da conexão
- Campo Sequence number
 - Indica o número de seqüência do primeiro byte de dados contido no seguimento



Protocolo TCP

■ Controle de seqüência

Números de seqüência:

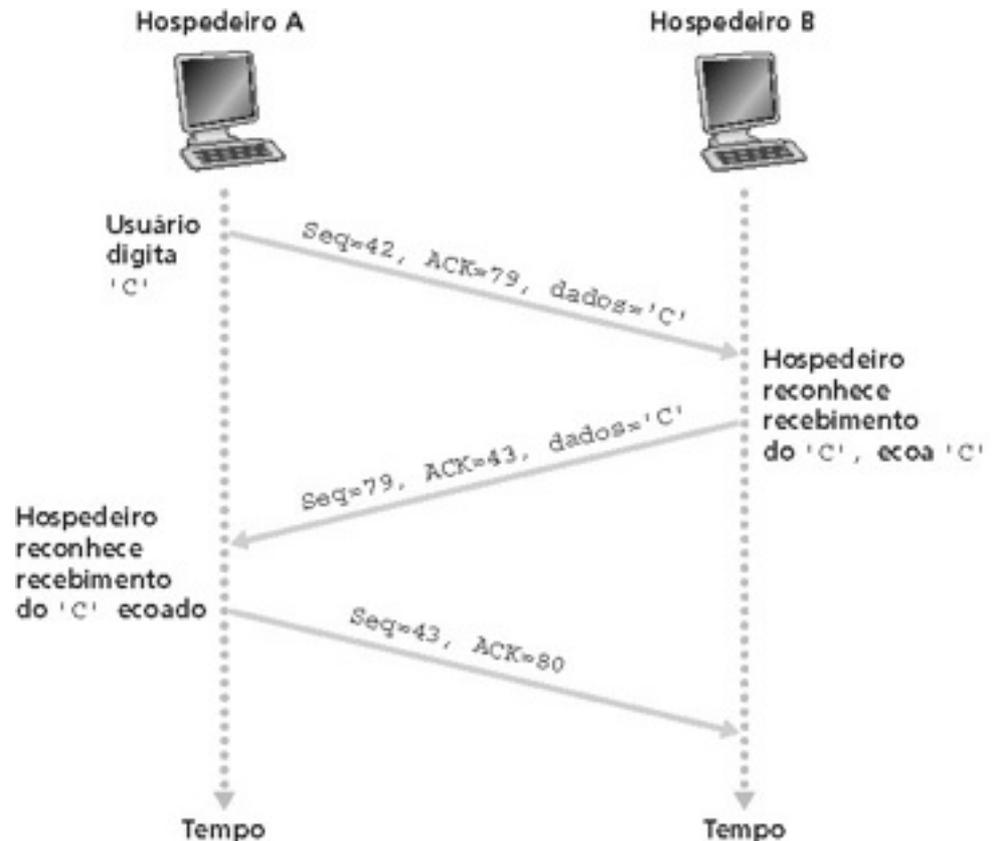
- Número do primeiro byte nos segmentos de dados

ACKs:

- Número do próximo byte esperado do outro lado
- ACK cumulativo

P.: Como o receptor trata segmentos fora de ordem?

- A especificação do TCP não define, fica a critério do implementador



Protocolo TCP

■ Controle de erros

■ Reconhecimento positivo

- Destino retorna uma mensagem indicando o correto recebimento do segmento
- Reconhecimento pode pegar carona no segmento de dados do fluxo inverso

■ Reconhecimento cumulativo

- Diversos segmentos consecutivos podem ser reconhecidos em uma única mensagem

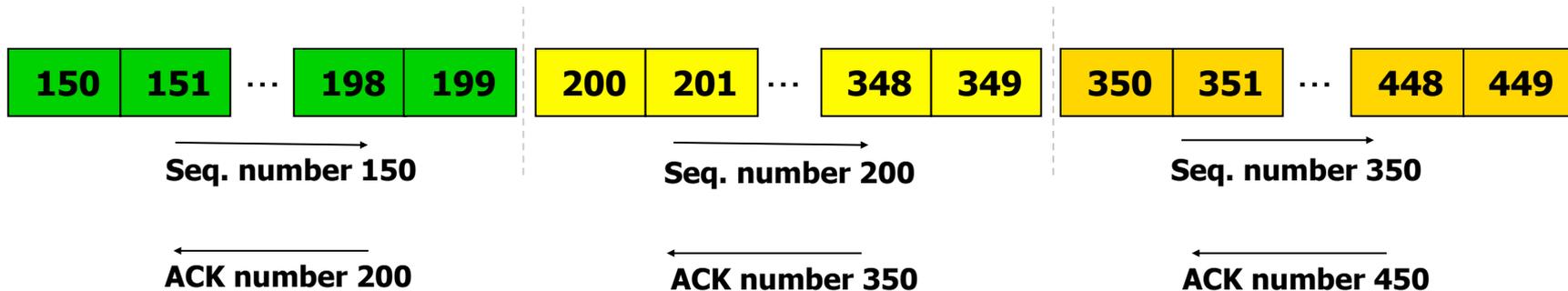


Protocolo TCP

■ Controle de erros

■ Acknowledgment number

- Indica o número de sequência do próximo byte que espera receber
- Indica o correto recebimento dos bytes com número de seqüência anterior
- Bit ACK do **Code Bits** deve ser ativado



Protocolo TCP

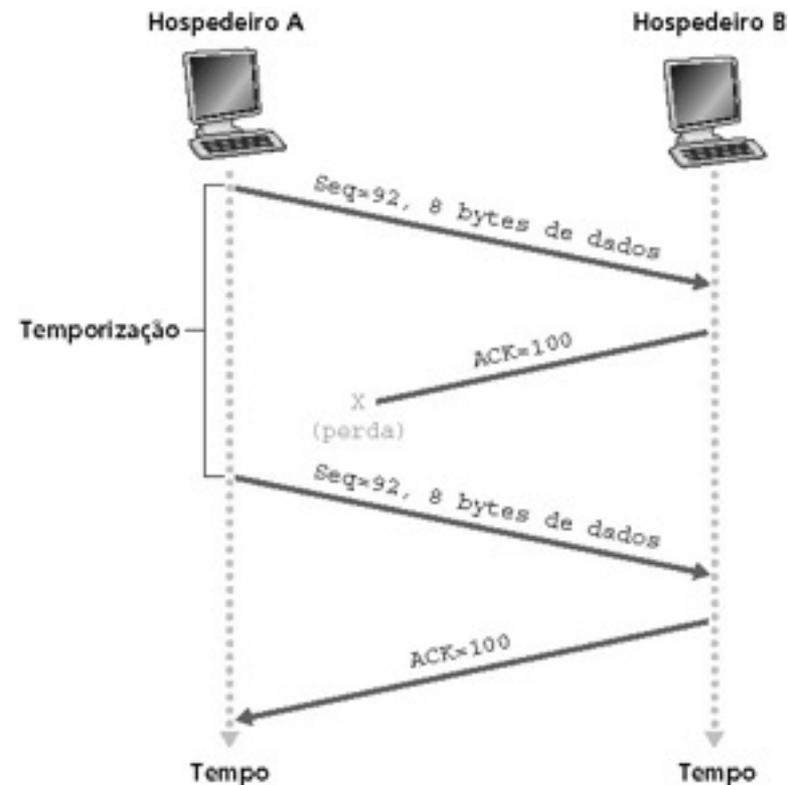
■ Controle de erros

■ Realizado através de Retransmissão

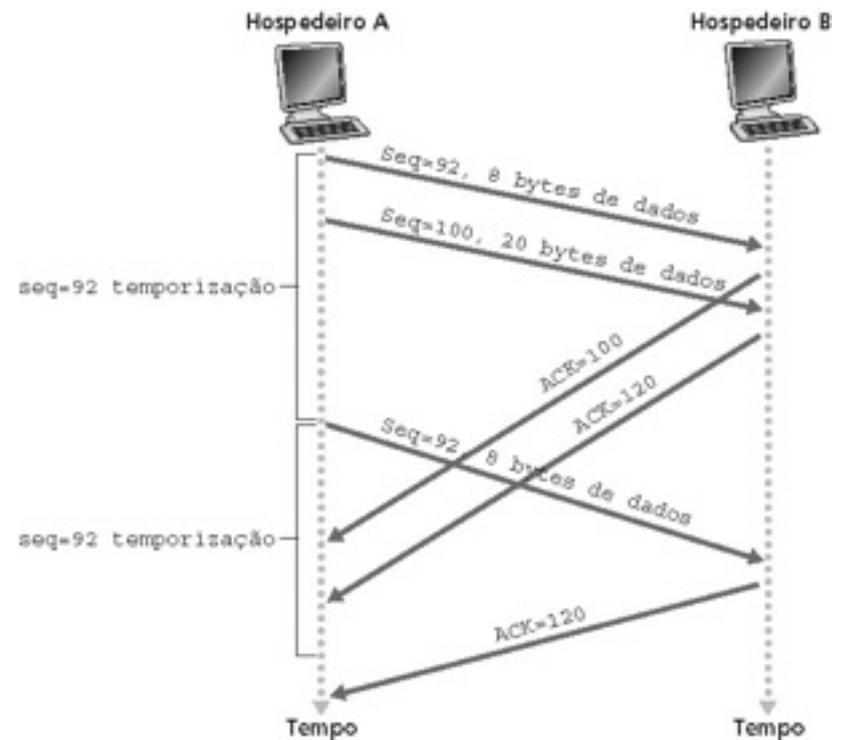
- Origem adota um temporizador para cada segmento enviado
- Segmento é retransmitido quando a origem não recebe o **reconhecimento** (ack) antes de expirar o temporizador
- Temporizador é reativado em cada retransmissão

Protocolo TCP

■ Controle de erros - Cenários



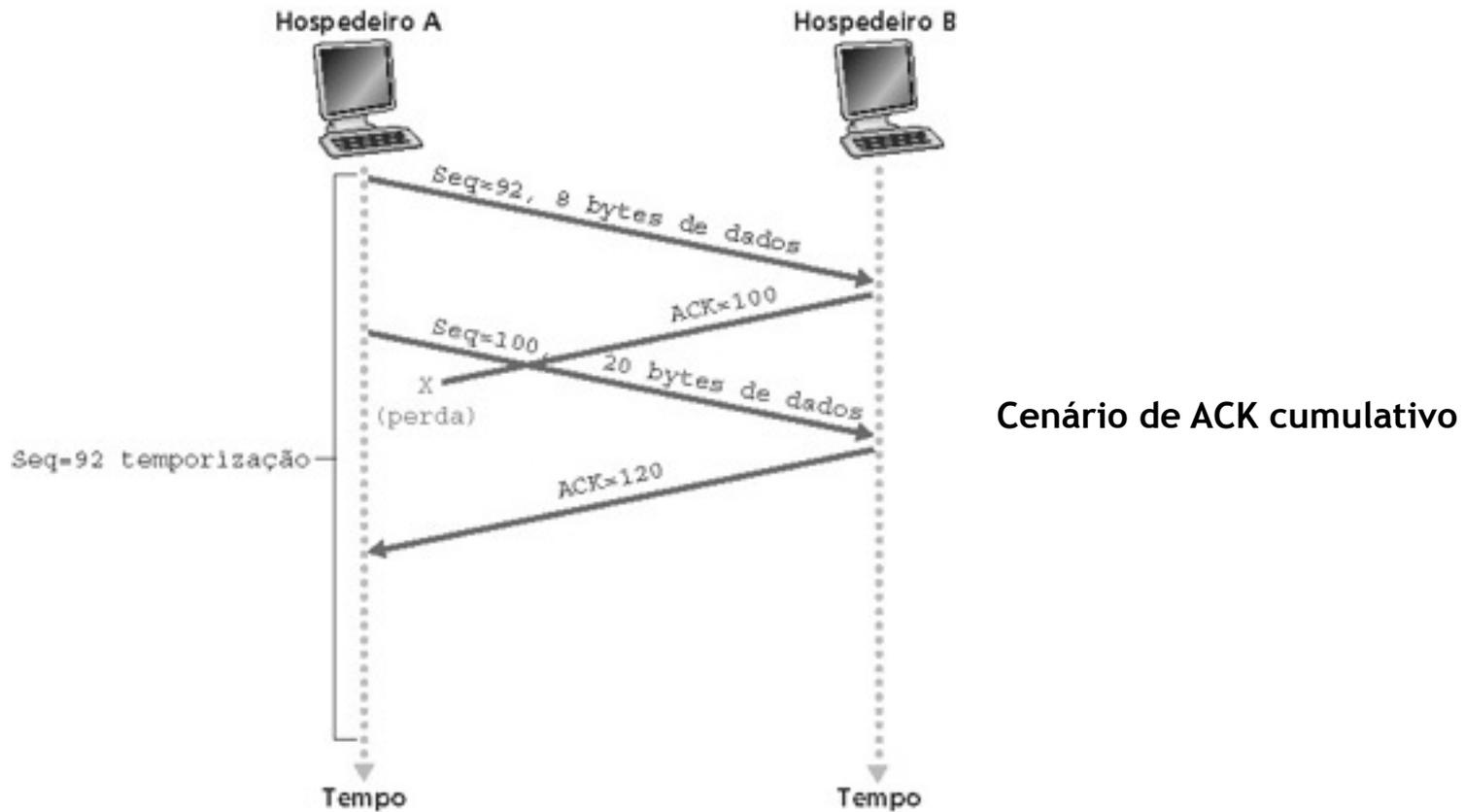
Cenário com perda
do ACK



Temporização prematura,
ACKs cumulativos

Protocolo TCP

■ Controle de erros - Cenários



Protocolo TCP

■ Controle de fluxo

■ Objetivo

- Transmissor não deve esgotar os buffers de recepção enviando dados rápido demais

■ Implementação

■ Janela deslizante

- Entidades negociam o número de bytes adicionais que podem ser recebidos a partir do último reconhecimento
- Destino define o tamanho de sua janela de recepção em cada segmento
- Origem atualiza o tamanho de sua janela de transmissão a cada reconhecimento
- Reconhecimento deslocam a janela de transmissão da origem para o primeiro byte sem reconhecimento

Protocolo TCP - Controle de fluxo

- lado receptor da conexão TCP possui um buffer de recepção:



- Processos de aplicação podem ser lentos para ler o buffer

Controle de fluxo

Transmissor não deve esgotar os buffers de recepção enviando dados rápido demais

- Serviço de **speed-matching**: encontra a taxa de envio adequada à taxa de vazão da aplicação receptora

Protocolo TCP

