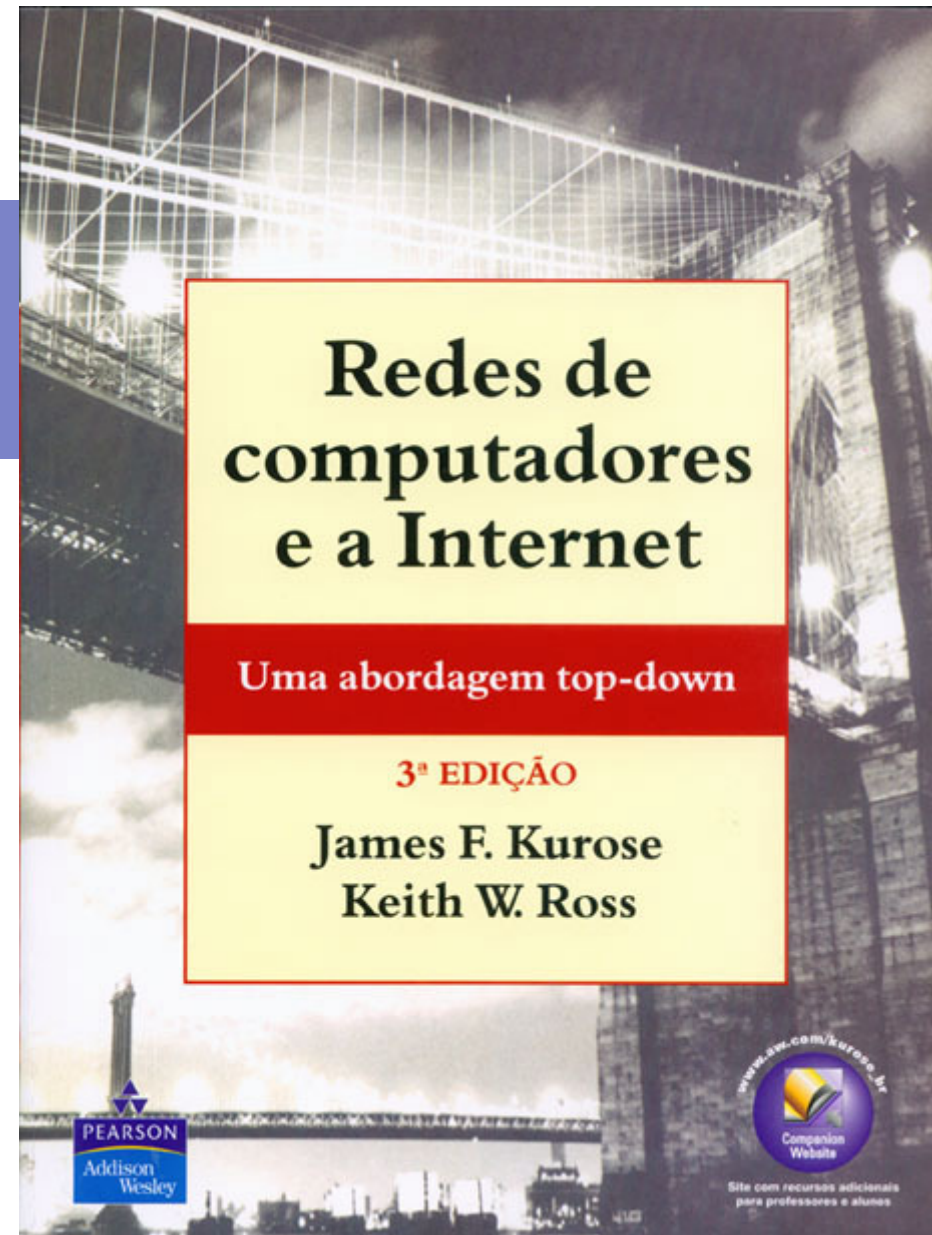


# Redes de computadores e a Internet

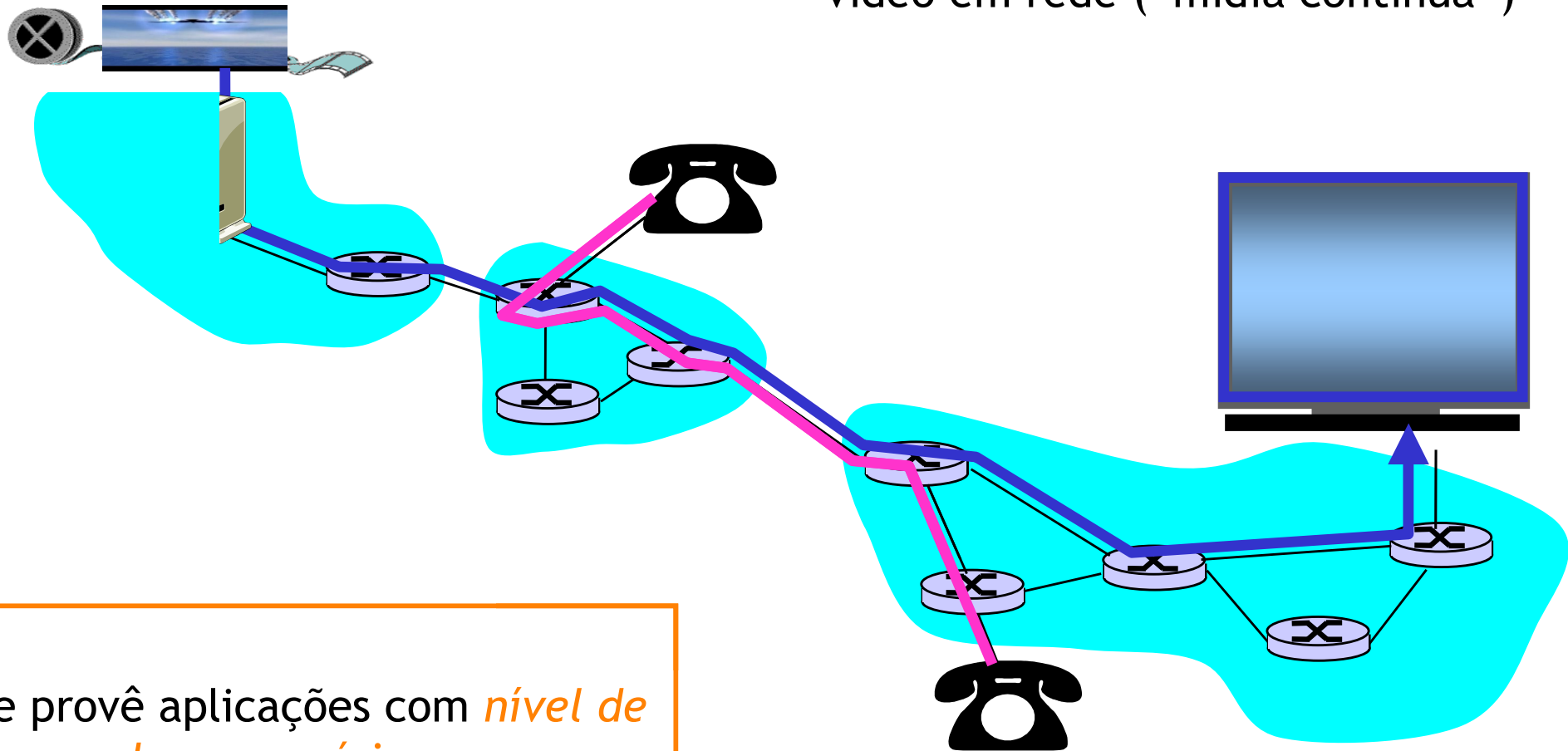
## Capítulo 7

**Redes  
multimídia**



# 7 Multimídia, qualidade de serviço: o que é?

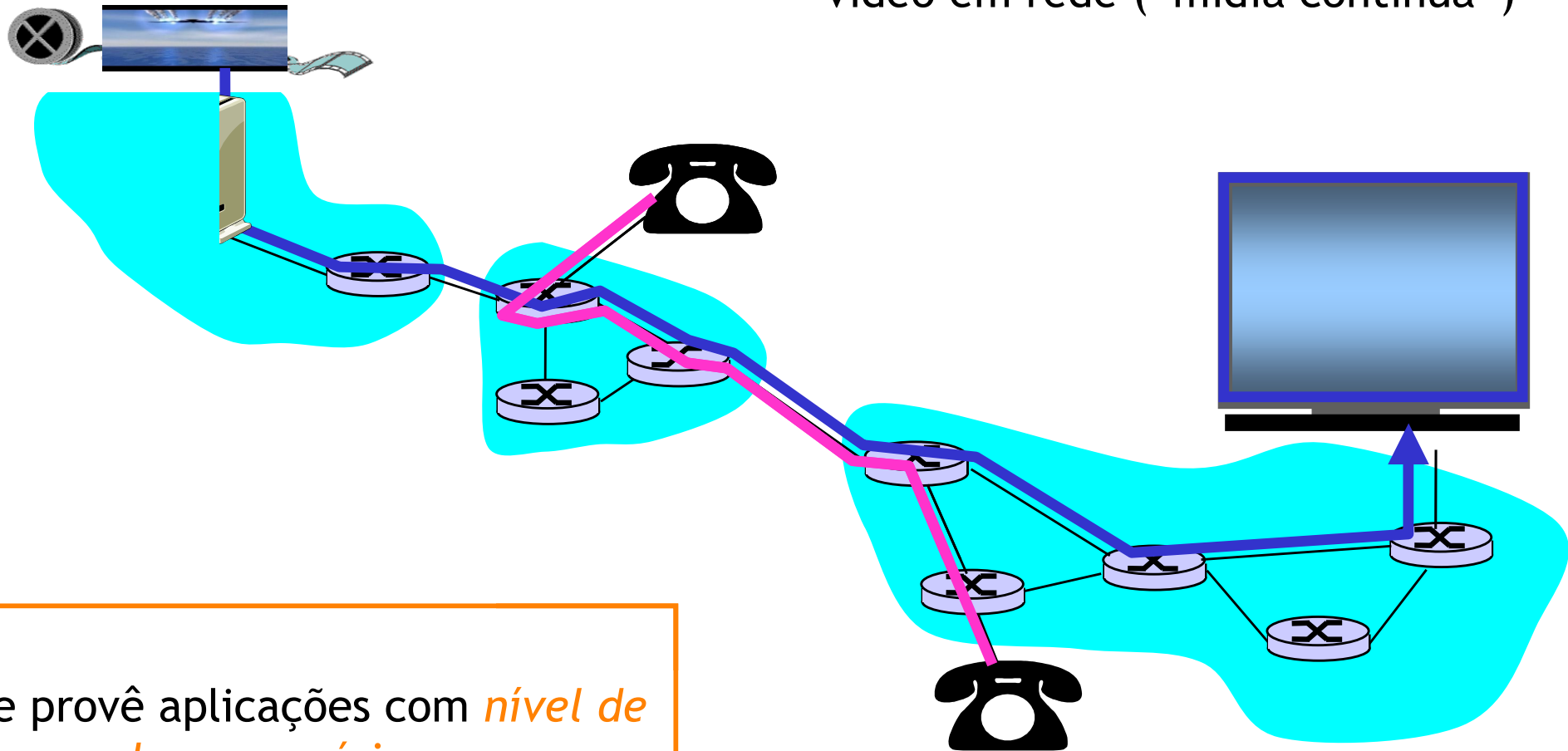
Aplicações multimídia: áudio e vídeo em rede (“mídia contínua”)



QoS  
rede provê aplicações com *nível de desempenho necessário para o funcionamento da aplicação.*

# 7 Multimídia, qualidade de serviço: o que é?

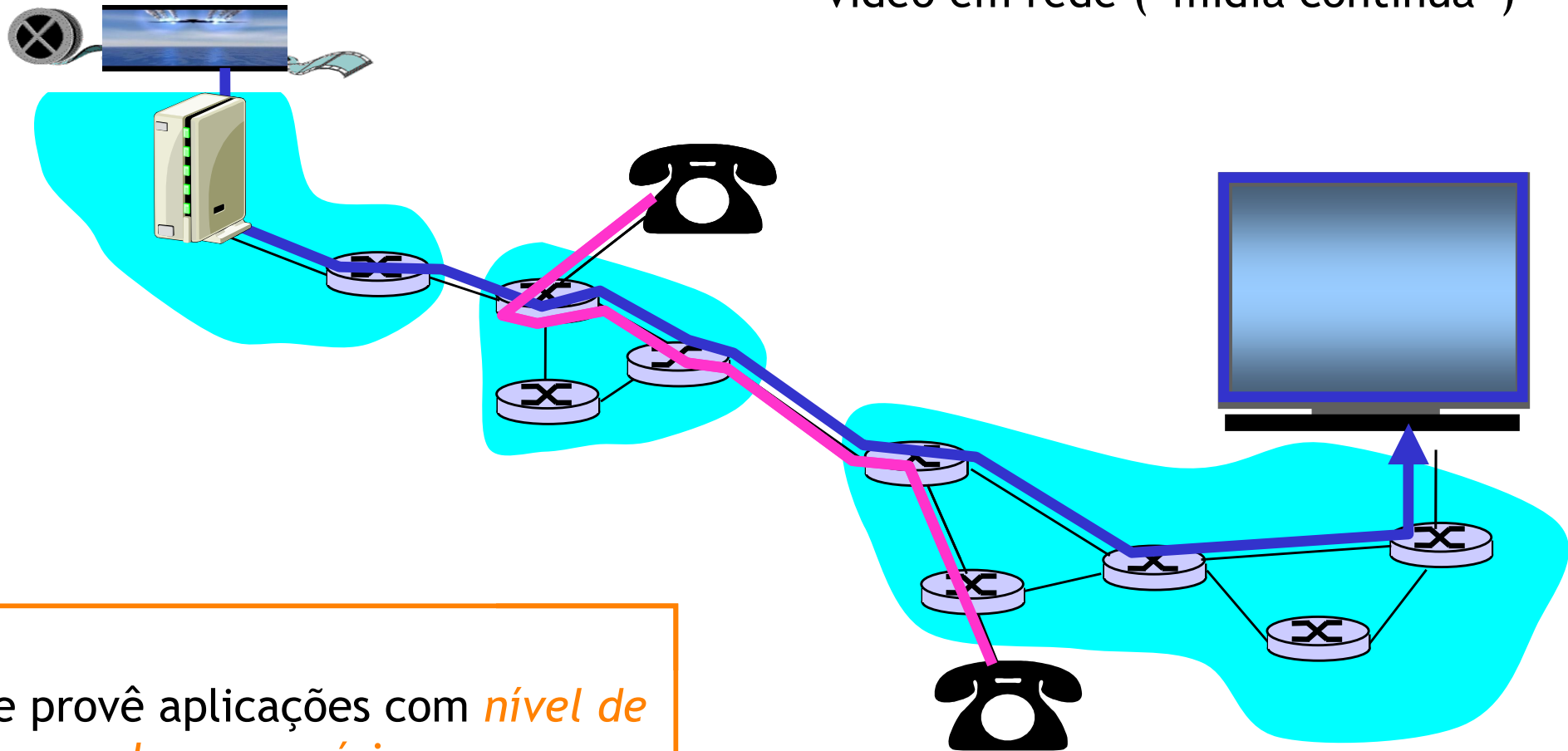
Aplicações multimídia: áudio e vídeo em rede (“mídia contínua”)



QoS  
rede provê aplicações com *nível de desempenho necessário para o funcionamento da aplicação.*

# 7 Multimídia, qualidade de serviço: o que é?

Aplicações multimídia: áudio e vídeo em rede (“mídia contínua”)



QoS  
rede provê aplicações com *nível de desempenho necessário para o funcionamento da aplicação.*

# 7 Redes multimídia

## Objetivos do capítulo

### Princípios

- Classificar aplicações multimídia
- Identificar os serviços de rede de que as aplicações necessitam
- Obtendo o melhor do serviço de melhor esforço
- Mecanismos para prover QoS

### Protocolos e arquiteturas

- Protocolos específicos para melhor esforço
- Arquiteturas para QoS

# 7 Redes multimídia

- 7.1 Aplicações de rede multimídia
- 7.2 Transmissão em fluxo contínuo de áudio e vídeo armazenados
- 7.3 Multimídia em tempo real: estudo da telefonia Internet
- 7.4 Protocolos para aplicações interativas em tempo real  
RTP, RTCP, SIP
- 7.5 Distribuição de Multimídia: redes de distribuição de conteúdo
- 7.6 Além do Melhor Esforço
- 7.7 Agendamento e Mecanismos de Policiamento
- 7.8 Serviços Integrados e Serviços Diferenciados
- 7.9 RSVP

# 7 Aplicações de rede MM

## Classes de aplicações MM:

- 1) Transmissão em fluxo contínuo de áudio e vídeo armazenados
- 2) Transmissão em fluxo contínuo de áudio e vídeo ao vivo
- 3) Áudio e vídeo interativos em tempo real

**Jitter é a variação de atrasos dos pacotes dentro do mesmo fluxo de pacotes**

## Características fundamentais:

- Tipicamente **sensível a atraso**
  - Atraso fim-a-fim
  - Jitter do atraso
- Mas **tolerante a perdas**: perdas infreqüentes causam interrupções aleatórias menores
- Antítese dos dados, que são intolerantes a perdas, mas tolerantes a atrasos.



PEARSON

Addison  
Wesley

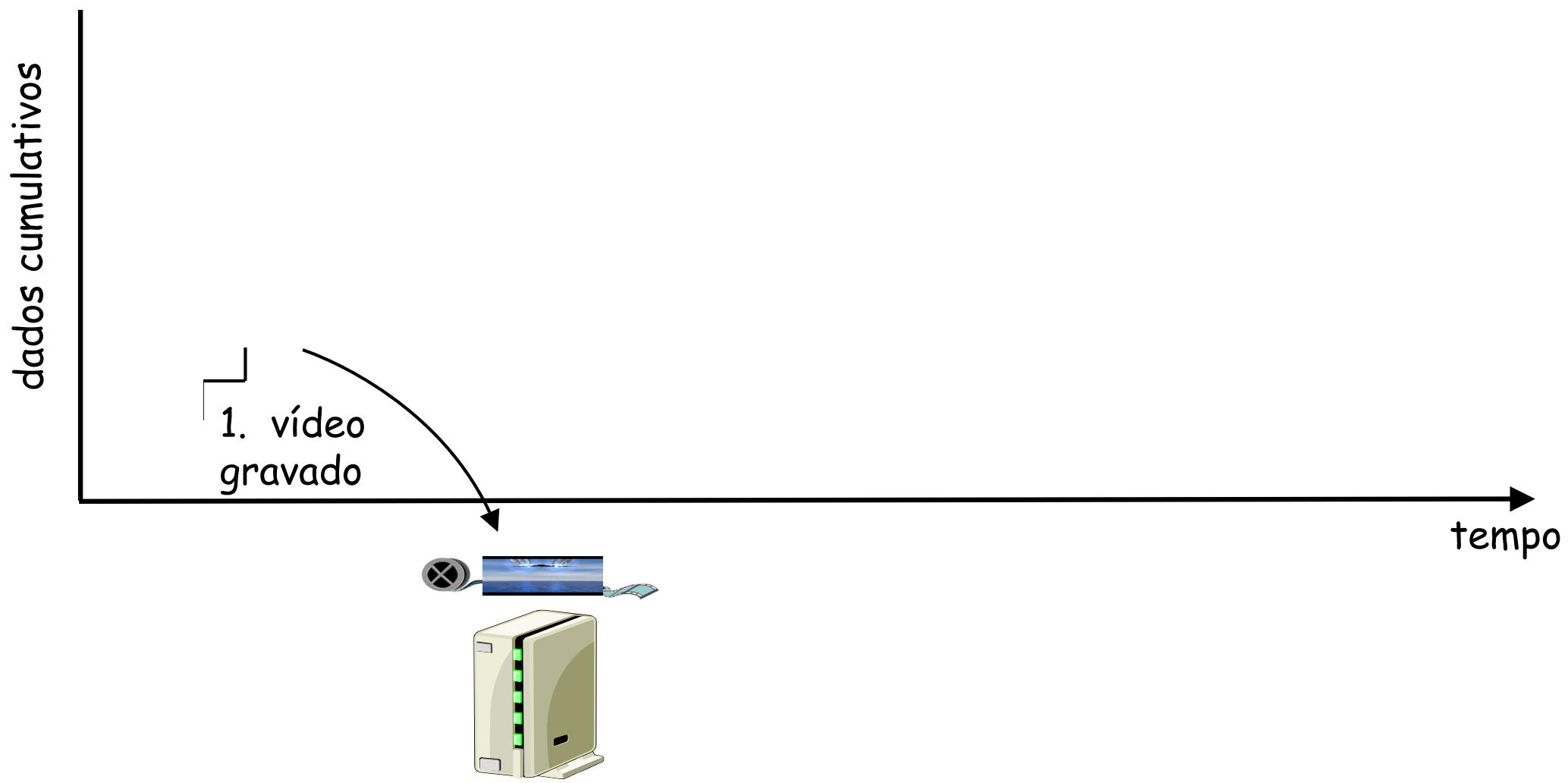
## Transmissão em fluxo contínuo:

- Mídia armazenada na origem
- Transmitida para o cliente
- Fluxo contínuo: cliente começa a reprodução *antes* que todos os dados tenham chegado
- Confinamento de tempo para os dados que ainda serão transmitidos: em tempo para a reprodução

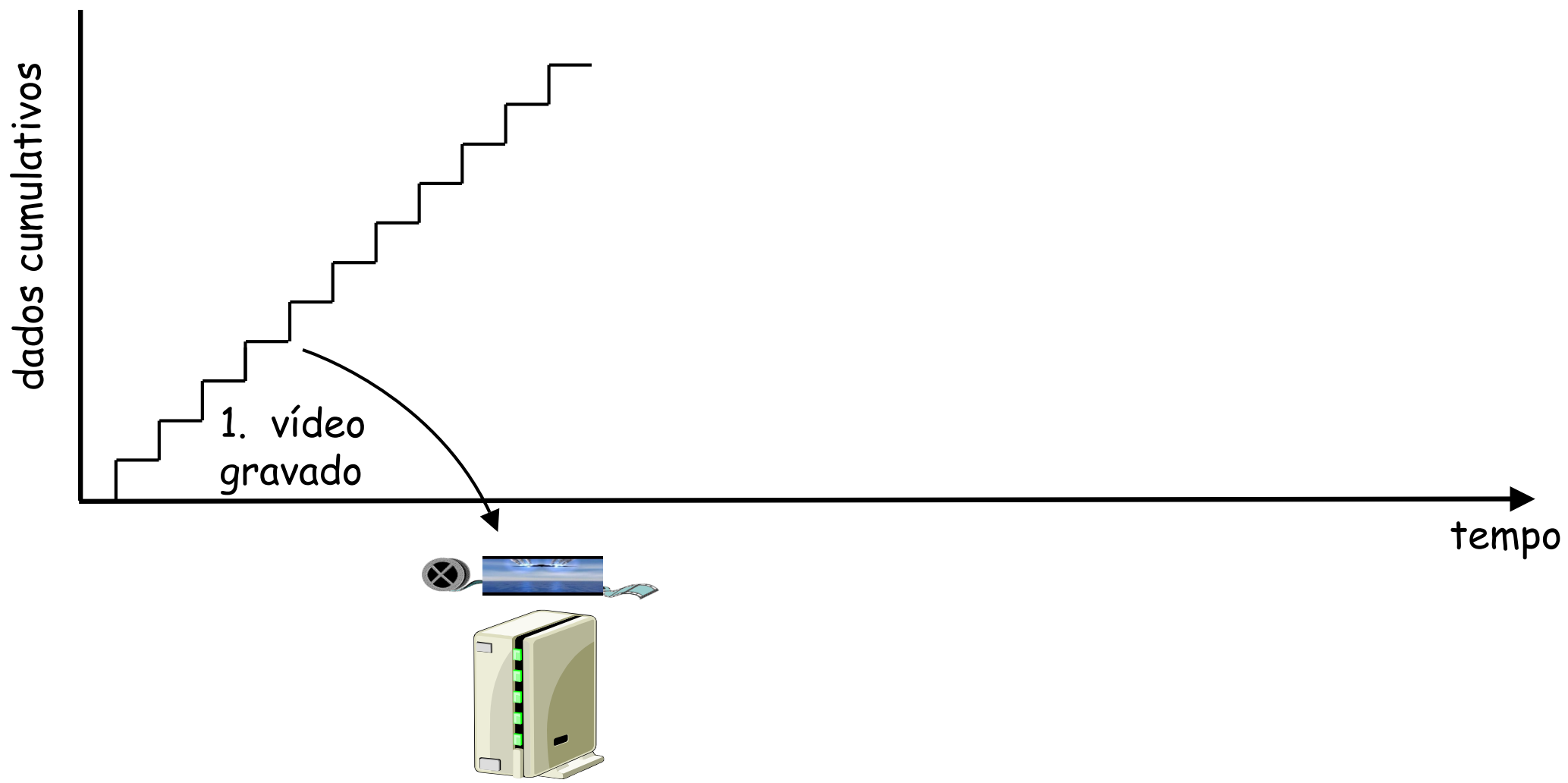




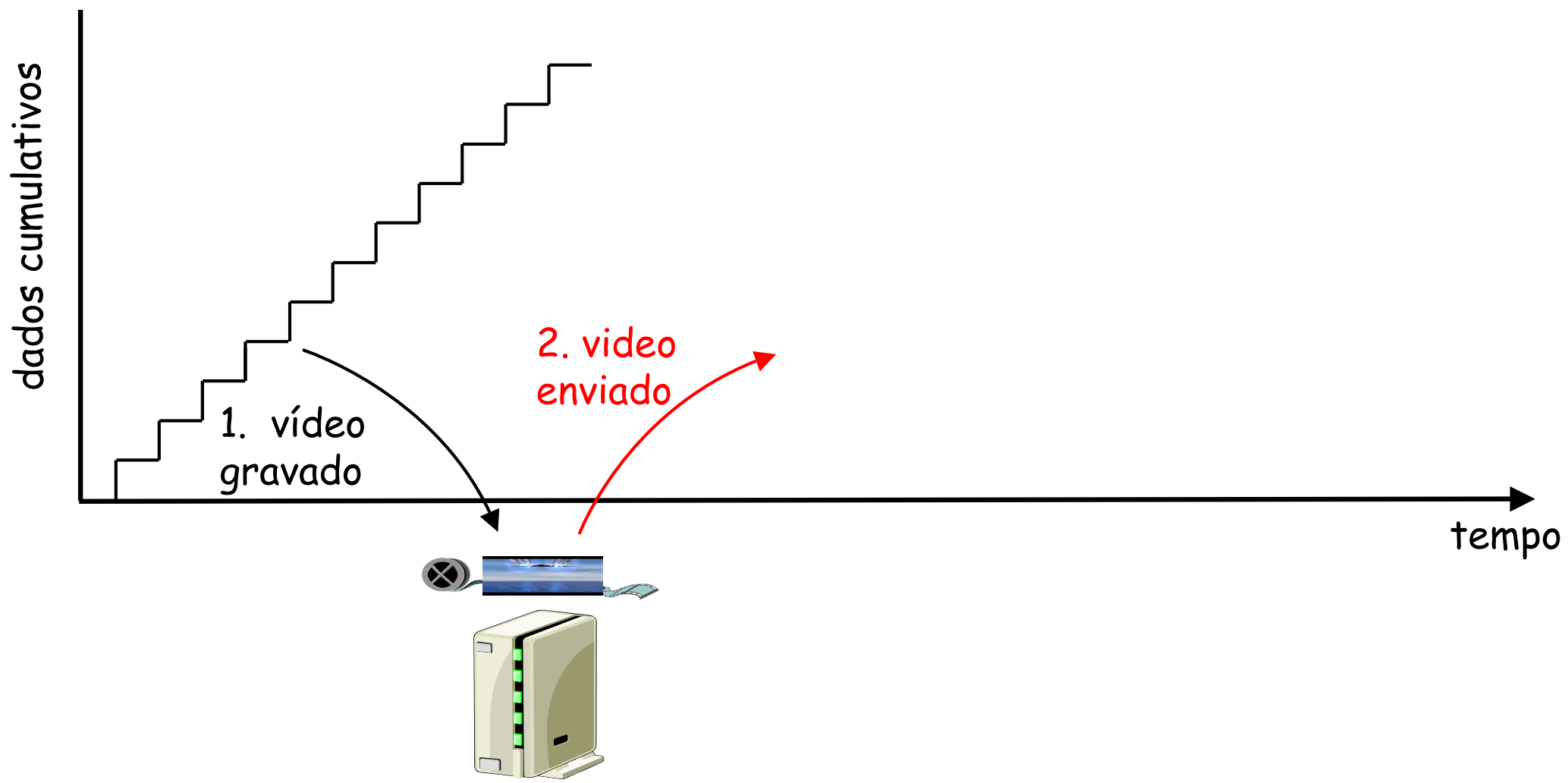
# 7 Transmissão em fluxo contínuo de multimídia armazenada: o que é?



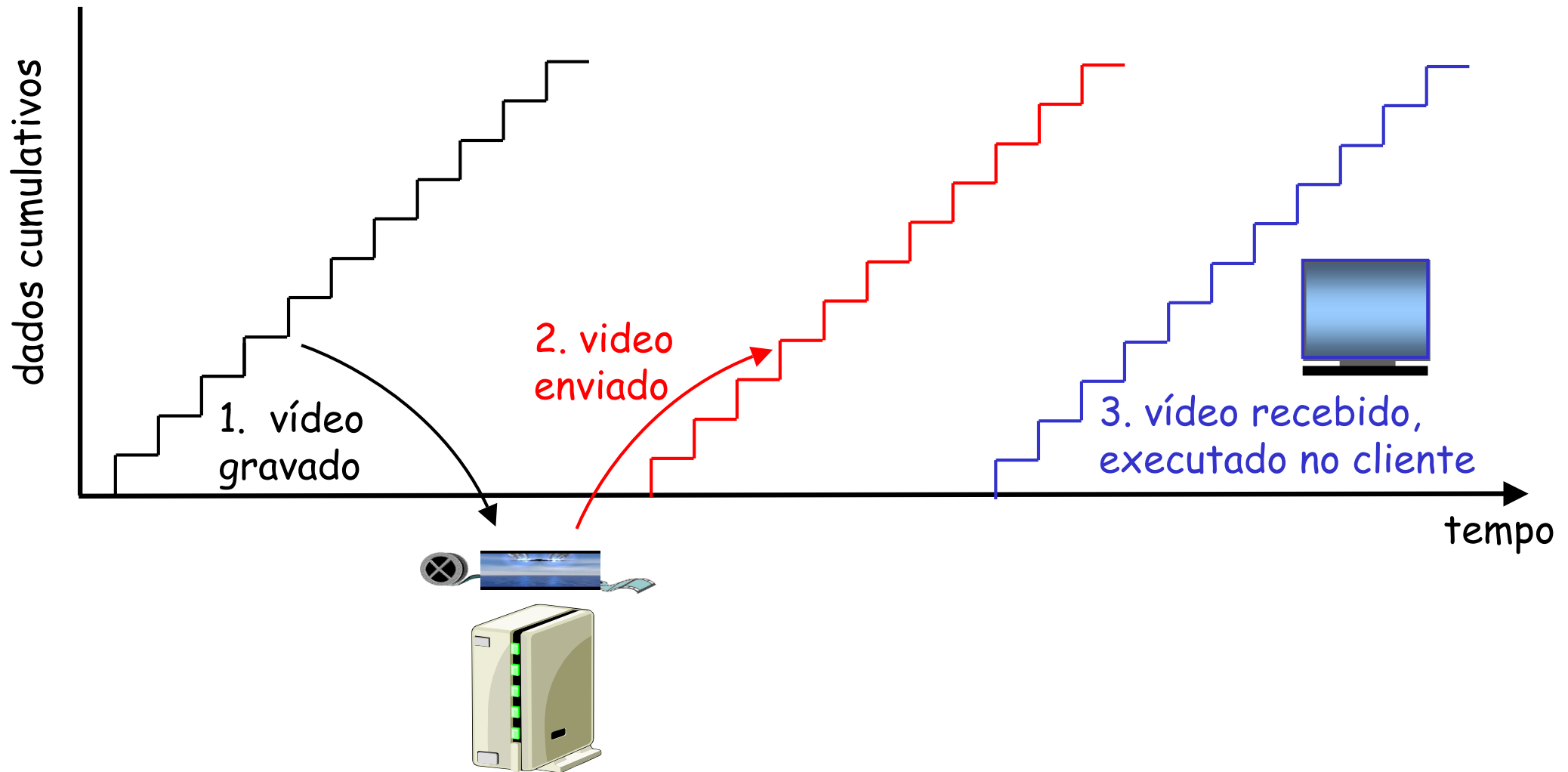
# 7 Transmissão em fluxo contínuo de multimídia armazenada: o que é?



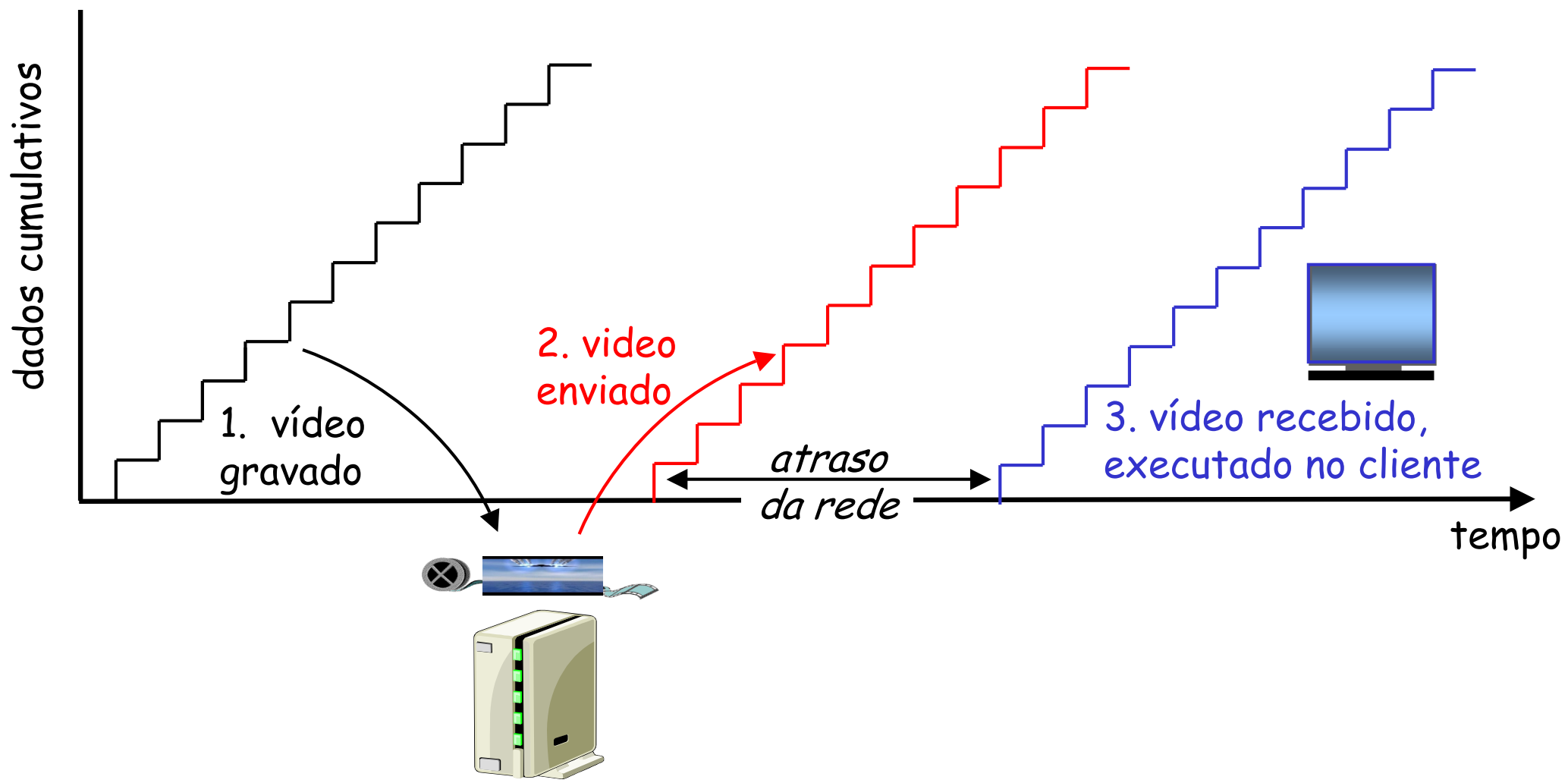
# 7 Transmissão em fluxo contínuo de multimídia armazenada: o que é?



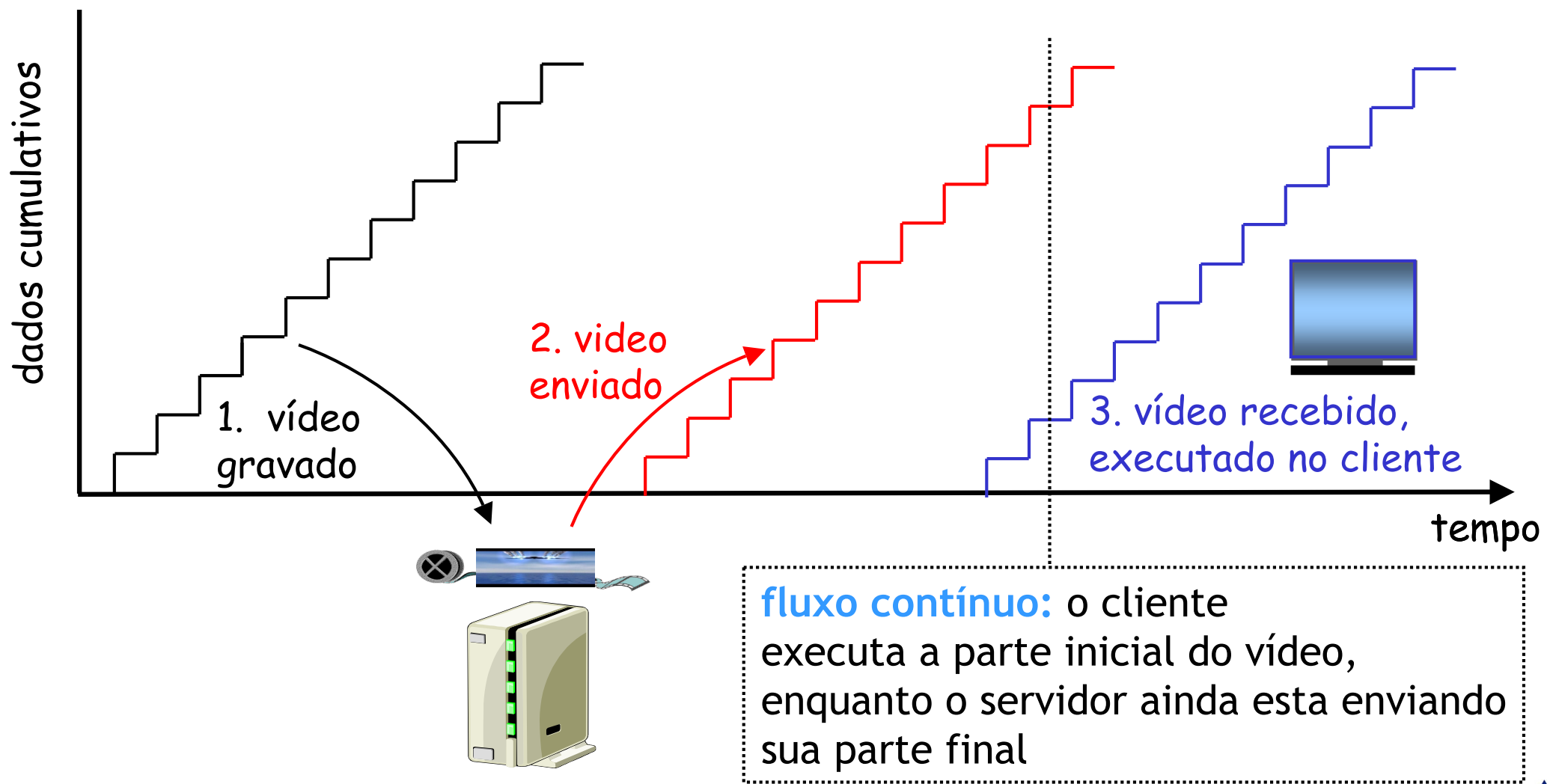
# 7 Transmissão em fluxo contínuo de multimídia armazenada: o que é?



# 7 Transmissão em fluxo contínuo de multimídia armazenada: o que é?



# 7 Transmissão em fluxo contínuo de multimídia armazenada: o que é?



# 7 Transmissão em fluxo contínuo de multimídia armazenada: interatividade

- **Funcionalidade igual ao VCR:** cliente pode parar, voltar, adiantar, mover a barra de transição
  - 10 segundos de atraso inicial OK
  - 1-2 segundos até efeito de comando OK
  - RTSP freqüentemente utilizado (mais tarde)
- Confinamento de tempo para os dados que ainda serão transmitidos: em tempo para a reprodução



## Exemplos:

- *Talk show* de rádio na Internet
- Evento esportivo ao vivo

## Fluxo contínuo

- Buffer de reprodução
- Reprodução pode retardar dez segundos após a transmissão
- Ainda possui confinamento de tempo

## Interatividade

- É impossível adiantar
- É possível voltar e parar!





# 7 Multimídia interativa em tempo real

- **Aplicações:** telefonia IP, videoconferência, mundos interativos distribuídos
- **Requisitos de atraso fim-a-fim:**
  - Áudio: < 150 mseg bom, < 400 mseg OK
  - Inclui atrasos do nível de aplicação (empacotamento) e de rede
  - Atrasos maiores notáveis, danificam a interatividade
- **Inicialização da sessão**
  - Como o chamador comunica seu endereço IP, número de porta, algoritmos de codificação?



# 7 Multimídia sobre a Internet hoje

TCP/UDP/IP: “serviço de melhor esforço”

- **nenhuma** garantia contra atrasos e perdas

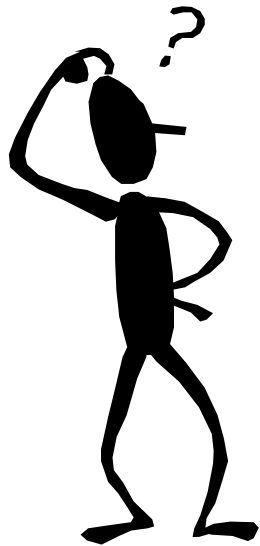
? ? ? ? ?  
Mas disseram que aplicações multimídia  
requerem QoS e nível de desempenho  
para serem efetivas!  
? ? ?

Aplicações multimídia da Internet de hoje  
usam técnicas de nível de aplicação para amenizar  
(o máximo possível) os efeitos de atrasos e perdas

# 7 Multimídia sobre a Internet hoje

TCP/UDP/IP: “serviço de melhor esforço”

- **nenhuma** garantia contra atrasos e perdas



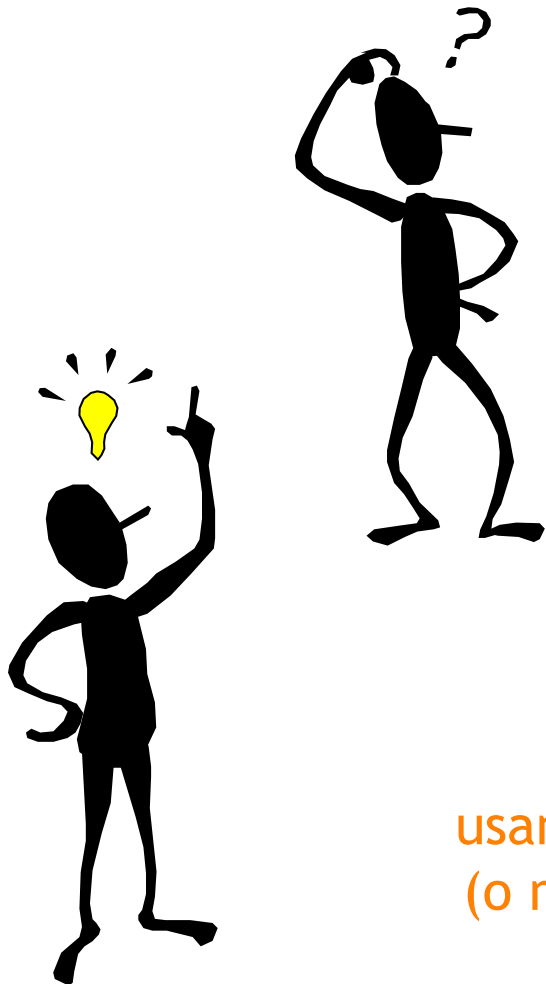
? ? ? ? ? ?  
Mas disseram que aplicações multimídia  
requerem QoS e nível de desempenho  
para serem efetivas!  
? ? ?

Aplicações multimídia da Internet de hoje  
usam técnicas de nível de aplicação para amenizar  
(o máximo possível) os efeitos de atrasos e perdas

# 7 Multimídia sobre a Internet hoje

TCP/UDP/IP: “serviço de melhor esforço”

- **nenhuma** garantia contra atrasos e perdas



? ? ? ? ?  
Mas disseram que aplicações multimídia  
requerem QoS e nível de desempenho  
para serem efetivas!  
? ? ?

Aplicações multimídia da Internet de hoje  
usam técnicas de nível de aplicação para amenizar  
(o máximo possível) os efeitos de atrasos e perdas

# 7 Como a Internet deveria evoluir para suportar melhor as aplicações multimídia?

## Filosofia de serviços integrados:

- Mudanças fundamentais na Internet para que as aplicações possam fazer reserva de banda fim-a-fim
- Requer softwares novos e complexos em hospedeiros e roteadores

## Não intervenção

- Sem maiores mudanças
- Maior largura de banda quando necessário
- Distribuição de conteúdo, multicast em nível de aplicação
  - Camada de aplicação

## Filosofia de serviços diferenciados:

- Poucas mudanças na infra-estrutura da Internet, ainda provê serviços de 1ª e 2ª classe.



Qual sua opinião?

- Sinal analógico amostrado numa taxa constante
  - Telefone: 8.000 amostras/s
  - Música de CD: 44.100 amostras/s
- Cada amostra é quantizada, isto é, arredondada
  - Ex.,  $2^8 = 256$  possíveis valores quantizados
- Cada valor quantizado é representado por bits
  - 8 bits para 256 valores
- Exemplo: 8.000 amostras/s, 256 valores quantizados --> 64.000 bps
- Receptor converte de volta para o sinal analógico:
  - Alguma redução de qualidade

## Exemplo de taxas

- CD: 1,411 Mbps
- MP3: 96, 128, 160 kbps
- telefonia Internet: 5,3 - 13 kbps



- Vídeo é uma seqüência de imagens mostradas numa taxa constante
  - ex., 24 imagens/s
- Imagem digital é um conjunto de pixels
- Cada pixel é representado por bits
- Redundância
  - Espacial
  - Temporal

## Exemplos:

- MPEG 1 (CD-ROM) 1.5 Mbps
- MPEG2 (DVD) 3-6 Mbps
- MPEG4 (freqüentemente usado em Internet, < 1 Mbps)

## Pesquisa:

- Vídeo em camadas (escalável)
  - Adapta as camadas para a largura de banda disponível

# 7 Redes multimídia

- 7.1 Aplicações de rede multimídia
- 7.2 Transmissão em fluxo contínuo de áudio e vídeo armazenados
- 7.3 Multimídia em tempo real: estudo da telefonia Internet
- 7.4 Protocolos para aplicações interativas em tempo real  
RTP, RTCP, SIP
- 7.5 Distribuição de Multimídia: redes de distribuição de conteúdo
- 7.6 Além do Melhor Esforço
- 7.7 Agendamento e Mecanismos de Policiamento
- 7.8 Serviços Integrados e Serviços Diferenciados
- 7.9 RSVP



Técnicas de fluxo contínuo em nível de aplicação para obter o melhor possível do serviço de melhor esforço:

- Buffer no lado cliente
- Uso de UDP versus TCP
- Múltiplas codificações de multimídia

## Media Player

- Remoção do jitter
- Descompressão
- Cancelamento de erro
- Interface gráfica de usuário com controles para interatividade

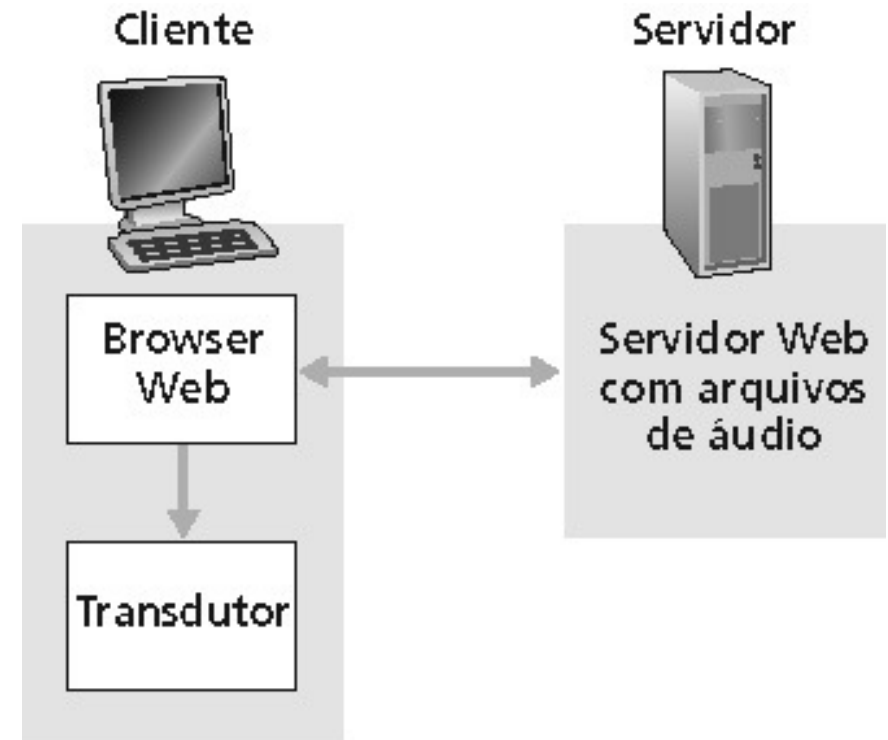
# 7

## Multimídia da Internet: abordagem mais simples

- Áudio ou vídeo armazenado em arquivo
- Arquivos transferidos como objeto HTTP
  - Recebidos por inteiro no cliente
  - Então iniciam a execução

### Áudio e vídeo sem fluxo contínuo:

- Sem “pipelining”, longos atrasos até a reprodução!



PEARSON

Addison  
Wesley

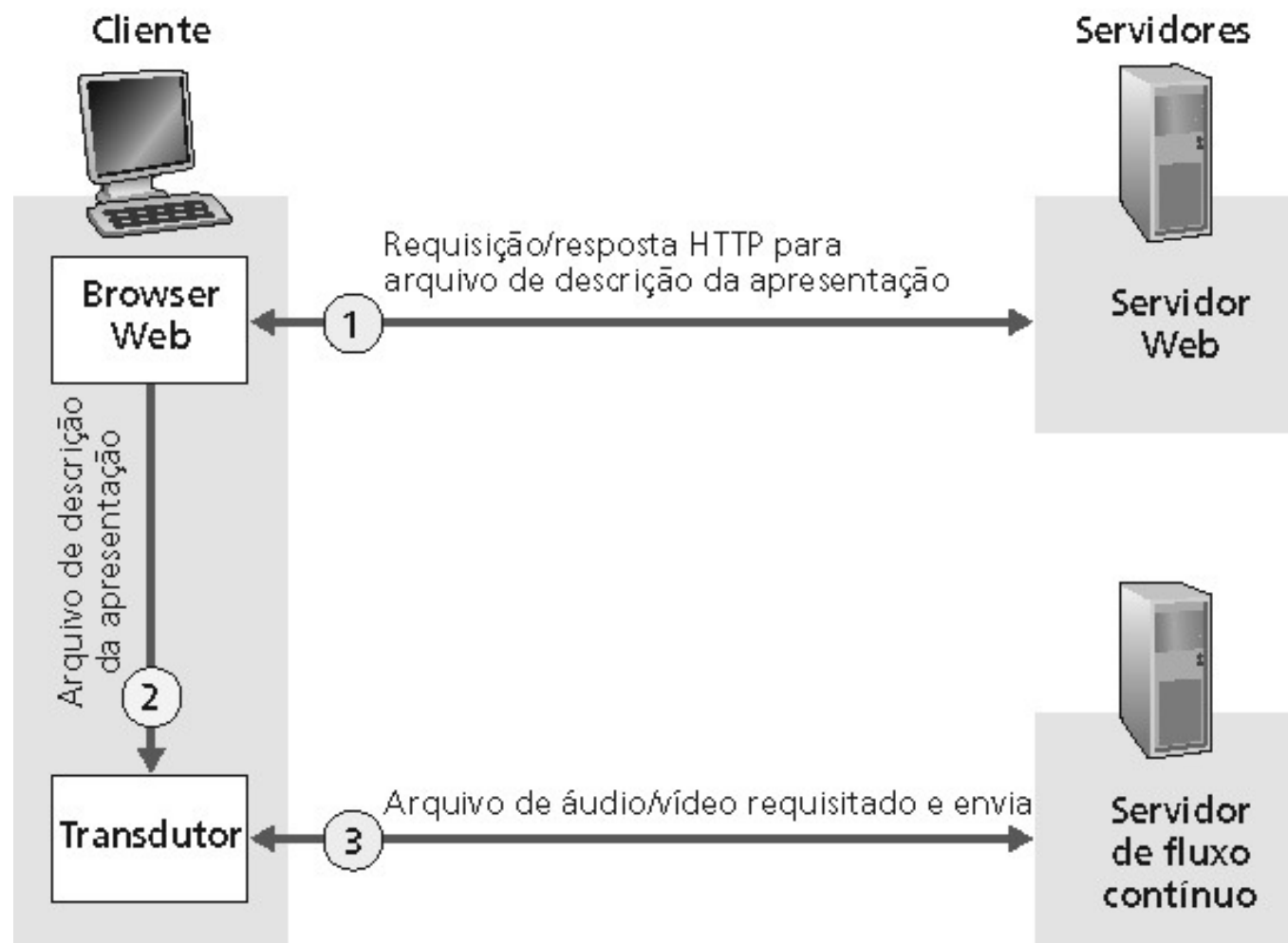
# 7

## Multimídia da Internet: abordagem de fluxo contínuo

- O browser obtém (GET) o **metarquivo**
- O browser aciona o player, passando o metarquivo
- O player contata o servidor
- O servidor transmite em **fluxo contínuo** o áudio/vídeo para o player



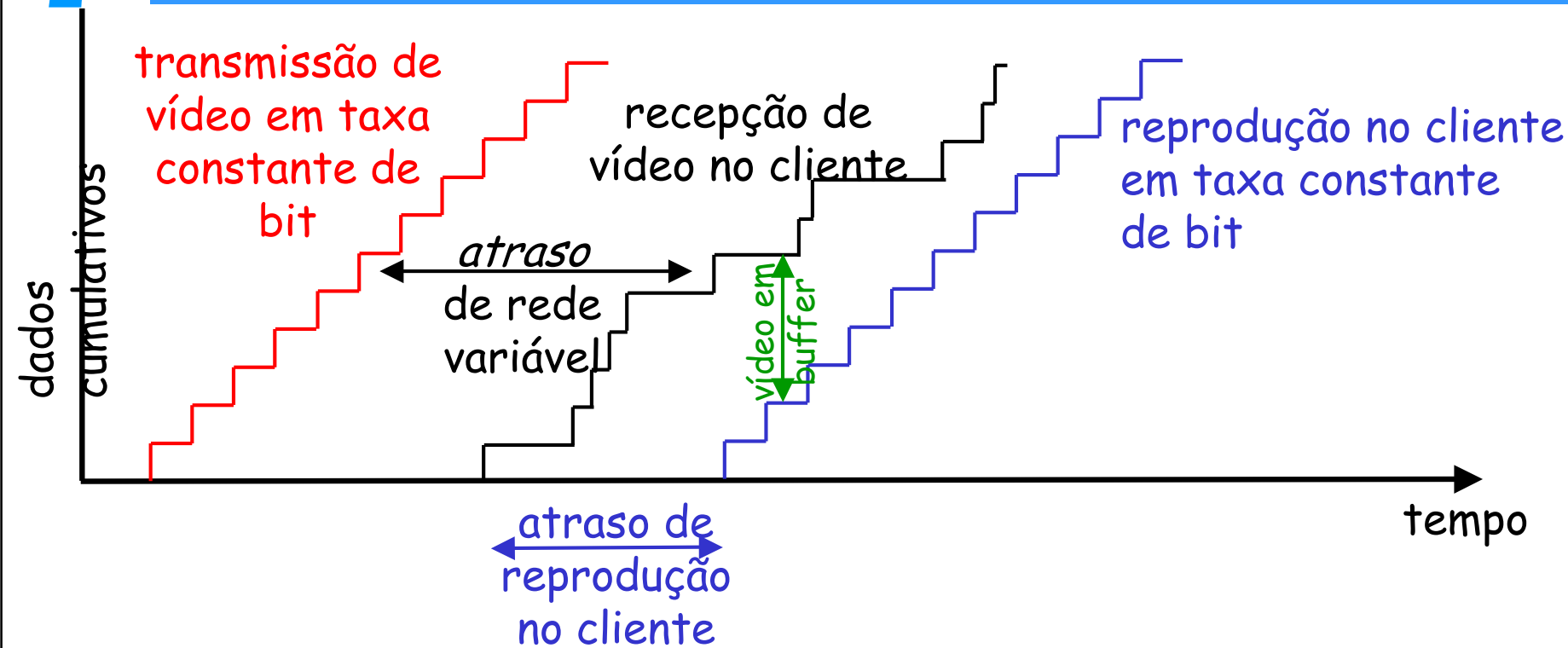
# 7 Transmitindo em fluxo contínuo a partir de um servidor de fluxo contínuo



- Esta arquitetura permite protocolo não-HTTP entre o servidor e o *media player*
- Também pode usar UDP em vez de TCP.

## 7

# Fluxo contínuo de multimídia: buffer no cliente



- Buffer no lado cliente, atraso de reprodução compensado pelo atraso adicionado pela rede, jitter

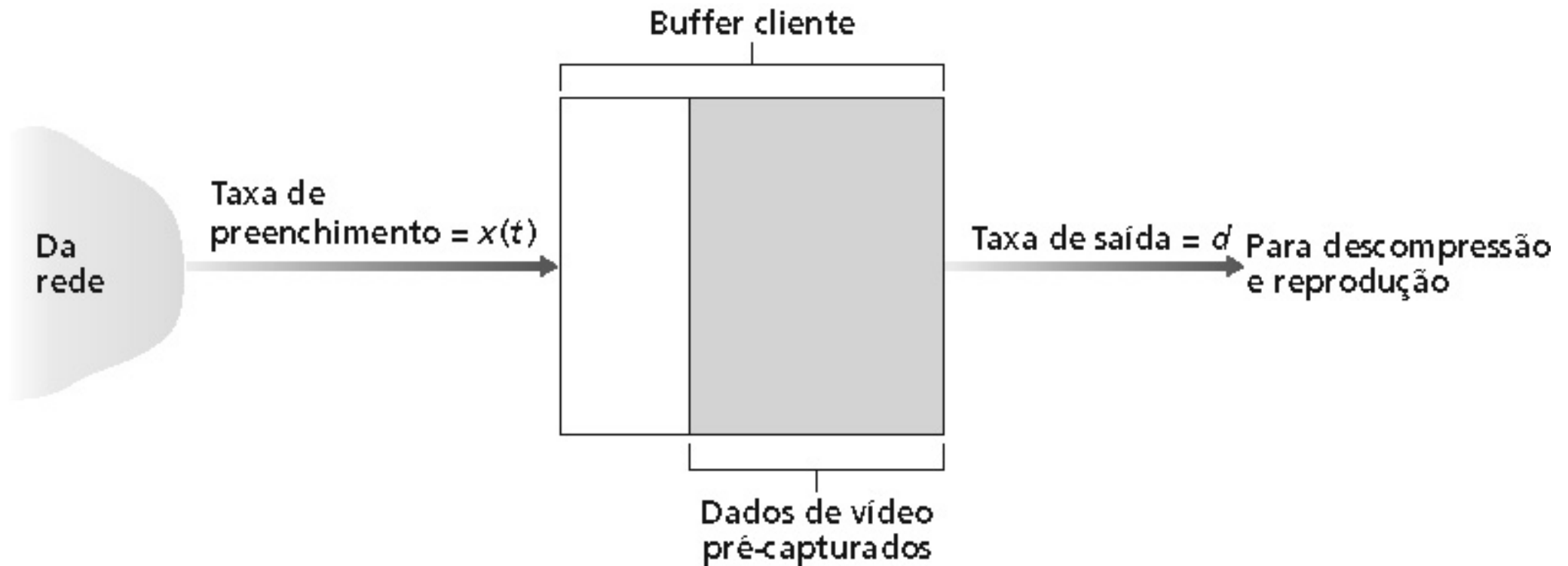


PEARSON

Addison  
Wesley

## 7

## Fluxo contínuo de multimídia: buffer no cliente



- Buffer no lado cliente, atraso de reprodução compensa o atraso adicionado pela rede, jitter



PEARSON

Addison  
Wesley

## UDP

- Servidor envia na taxa apropriada para o cliente (indiferente ao congestionamento da rede!)
  - Taxa de envio freqüente = taxa de codificação = taxa constante
  - Então, taxa de chegada = taxa constante - perda de pacotes
- Pequeno atraso de reprodução (2~5 s) para compensar o atraso de jitter da rede
- Recuperação de erros: permitido pelo tempo

## TCP

- Envia na máxima taxa possível sobre TCP
- Taxa de chegada flutua devido ao controle do congestionamento do TCP
- Maior atraso de execução: suaviza a taxa de entrega do TCP
- HTTP/TCP passa mais facilmente através dos firewalls



# 7

## Fluxo contínuo de multimídia: taxa(s) do cliente

**P.:** Como manipular diferentes capacidades de taxa de recepção do cliente?

- 28.8 kbps dialup
- 100 Mbps Ethernet

**R.:** servidor armazena, transmite múltiplas cópias de vídeo, codificado em taxas diferentes



PEARSON

Addison  
Wesley



## HTTP

- Não faz índice do conteúdo multimídia
- Sem comandos para avançar etc.

## RTSP: RFC 2326

- Protocolo da camada de aplicação cliente-servidor.
- Permite ao usuário controlar a apresentação: voltar ao início, avançar, parar, continuar etc. ...

## O que ele não faz:

- Não define como o áudio e o vídeo são encapsulados para transmissão sobre a rede
- Não restringe como a mídia contínua é transportada: pode usar UDP ou TCP
- Não especifica como o receptor armazena o áudio e o vídeo

# 7 RTSP: controle fora da banda

FTP usa um canal de controle “fora da banda”:

- Um arquivo é transferido sobre um canal
- Informação de controle (mudanças de diretório, remoção de arquivos, trocas de nomes etc.) é enviada sobre uma conexão TCP separada
- Os canais “dentro da banda” e “fora da banda” usam diferentes números de portas

Mensagens RTSP também são enviadas “fora da banda”:

- As mensagens de controle RTSP usam diferentes números de portas em relação ao fluxo de dados de mídia contínua, e, portanto, são enviados “fora da banda”.
  - Porta 554
- O fluxo contínuo de mídia é considerado “dentro da banda”.



# 7 RTSP: exemplo

## Cenário:

- Metarquivo comunicado ao web browser
- Browser aciona o player
- Player estabelece uma conexão de controle RTSP, conexão de dados ao servidor de fluxo contínuo

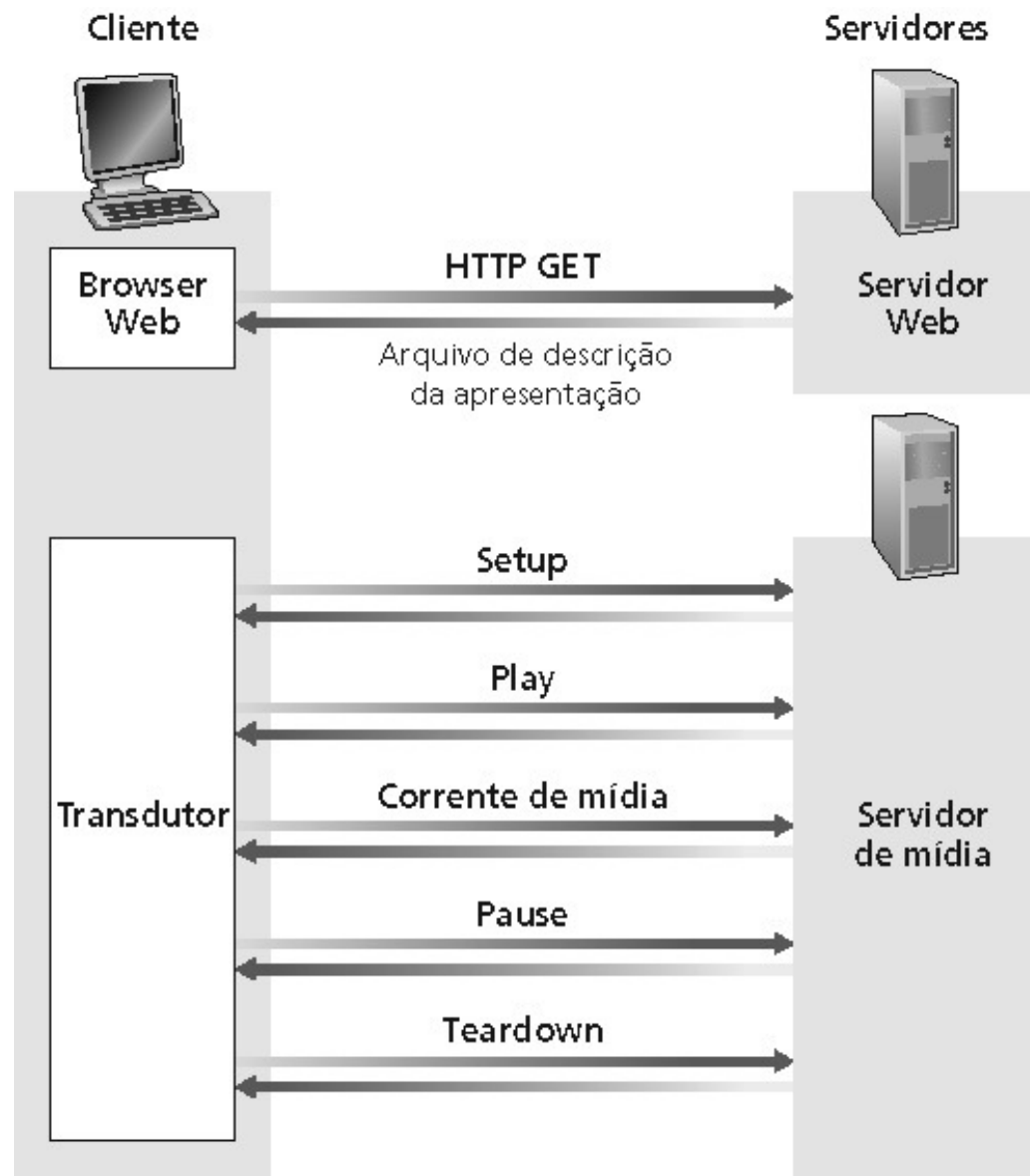
# 7

## Exemplo de metarquivo

```
<title>Twister</title>
<session>
  <group language=en lipsync>
    <switch>
      <track type=audio
        e="PCMU/8000/1"
        src = "rtsp://audio.example.com/twister/audio.en/lofi">
      <track type=audio
        e="DVI4/16000/2" pt="90 DVI4/8000/1"
        src="rtsp://audio.example.com/twister/audio.en/hifi">
    </switch>
  <track type="video/jpeg"
    src="rtsp://video.example.com/twister/video">
  </group>
</session>
```



# 7 RTSP: Operação



# 7

## Exemplo de troca RTSP

C: SETUP rtsp://audio.example.com/twister/audio RTSP/1.0  
Transport: rtp/udp; compression; port=3056; mode=PLAY

S: RTSP/1.0 200 1 OK  
Session 4231

C: PLAY rtsp://audio.example.com/twister/audio.en/lofi RTSP/1.0  
Session: 4231  
Range: npt=0-

C: PAUSE rtsp://audio.example.com/twister/audio.en/lofi RTSP/1.0  
Session: 4231  
Range: npt=37

C: TEARDOWN rtsp://audio.example.com/twister/audio.en/lofi RTSP/1.0  
Session: 4231

S: 200 3 OK

# 7 Redes multimídia

- 7.1 Aplicações de Rede Multimídia
- 7.2 Transmissão em fluxo contínuo de áudio e video armazenados
- 7.3 Multimídia em tempo real: estudo da Telefonia Internet
- 7.4 Protocolos para Aplicações Interativas em tempo real
  - RTP, RTCP, SIP
- 7.5 Distribuição de Multimídia: redes de distribuição de conteúdo
- 7.6 Além do Melhor Esforço
- 7.7 Agendamento e Mecanismos de Policiamento
- 7.8 Serviços Integrados e Serviços Diferenciados
- 7.9 RSVP

# 7

## Aplicações interativas em tempo real

- Telefone PC-a-PC
  - Serviços de mensagens instantâneas estão provendo isso
- PC-a-telefone
  - Dialpad
  - Net2phone
- Videoconferência com Webcams

Agora vamos examinar detalhadamente um exemplo de telefone PC-a-PC da Internet



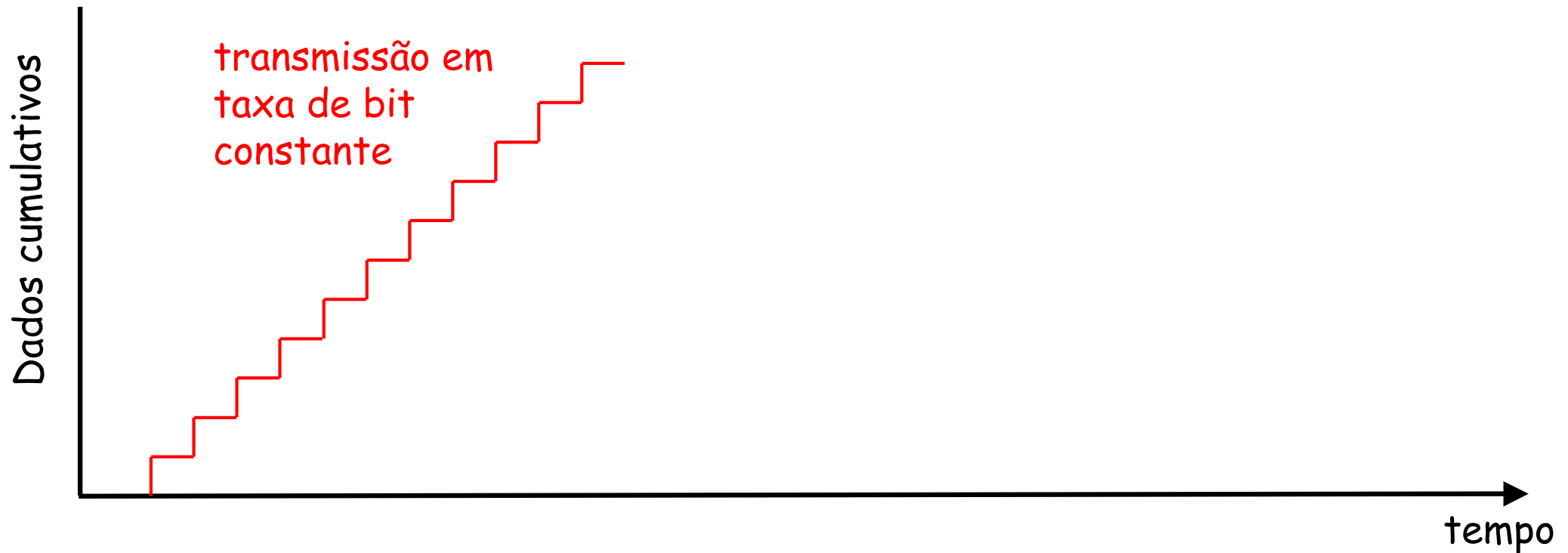
## Apresentamos a telefonia Internet através de um exemplo

- Áudio do orador: alterna períodos de fala e períodos de silêncio
  - 64 kbps durante o intervalo de atividade
- Pacotes gerados apenas durante períodos de fala
  - Blocos de 20 mseg a 8 Kbytes/s: dados de 160 bytes
- Cabeçalho da camada de aplicação adicionado a cada bloco
- Bloco + cabeçalhos encapsulados dentro do segmento UDP
- Aplicação envia o segmento UDP para o socket a cada 20 mseg durante o intervalo de atividade



- **Perda pela rede:** datagrama IP é perdido devido ao congestionamento da rede (sobrecarga do buffer do roteador)
- **Perda por atraso:** datagrama IP chega muito atrasado para a reprodução no receptor
  - Atrasos: processamento, fila na rede; atrasos no sistema final (transmissor, receptor)
  - Máxima tolerância de atraso: 400 ms
- Tolerância de perda: dependendo da codificação de voz, as perdas ficam ocultas, taxas de perda de pacotes entre 1% e 10% podem ser toleradas.

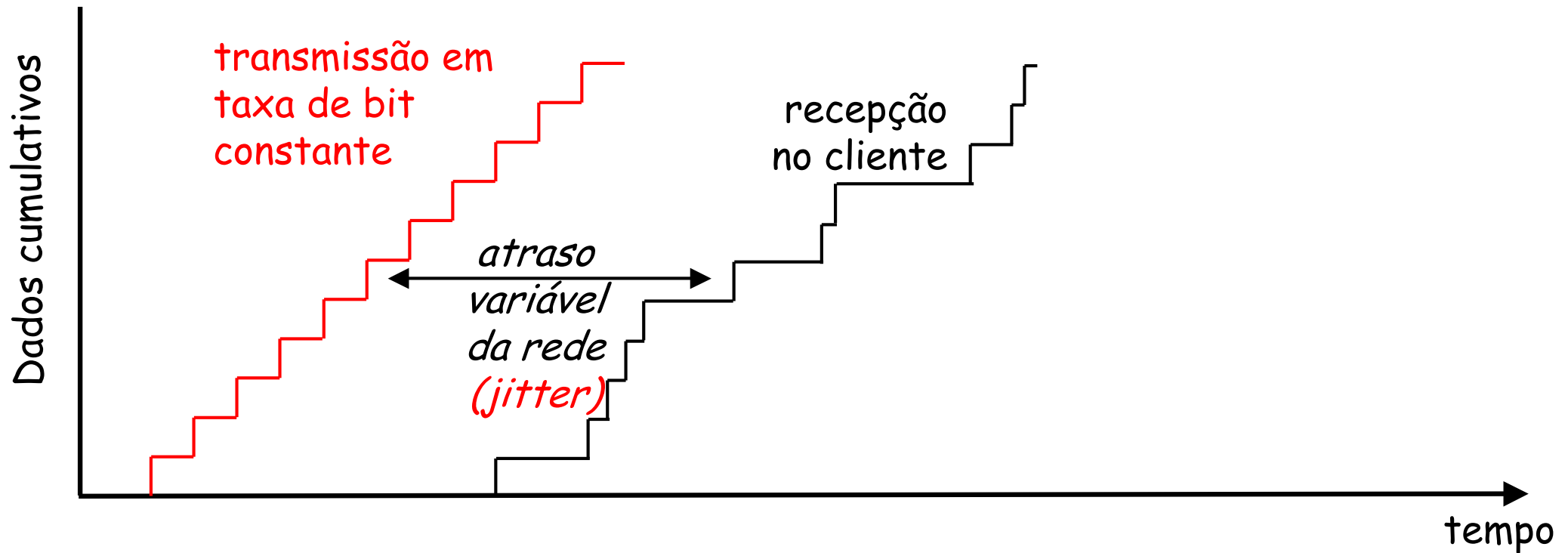




- Considere os atrasos fim-a-fim de dois pacotes consecutivos: diferença pode ser mais ou menos do que 20 mseg

## 7

## Jitter



- Considere os atrasos fim-a-fim de dois pacotes consecutivos: diferença pode ser mais ou menos do que 20 mseg

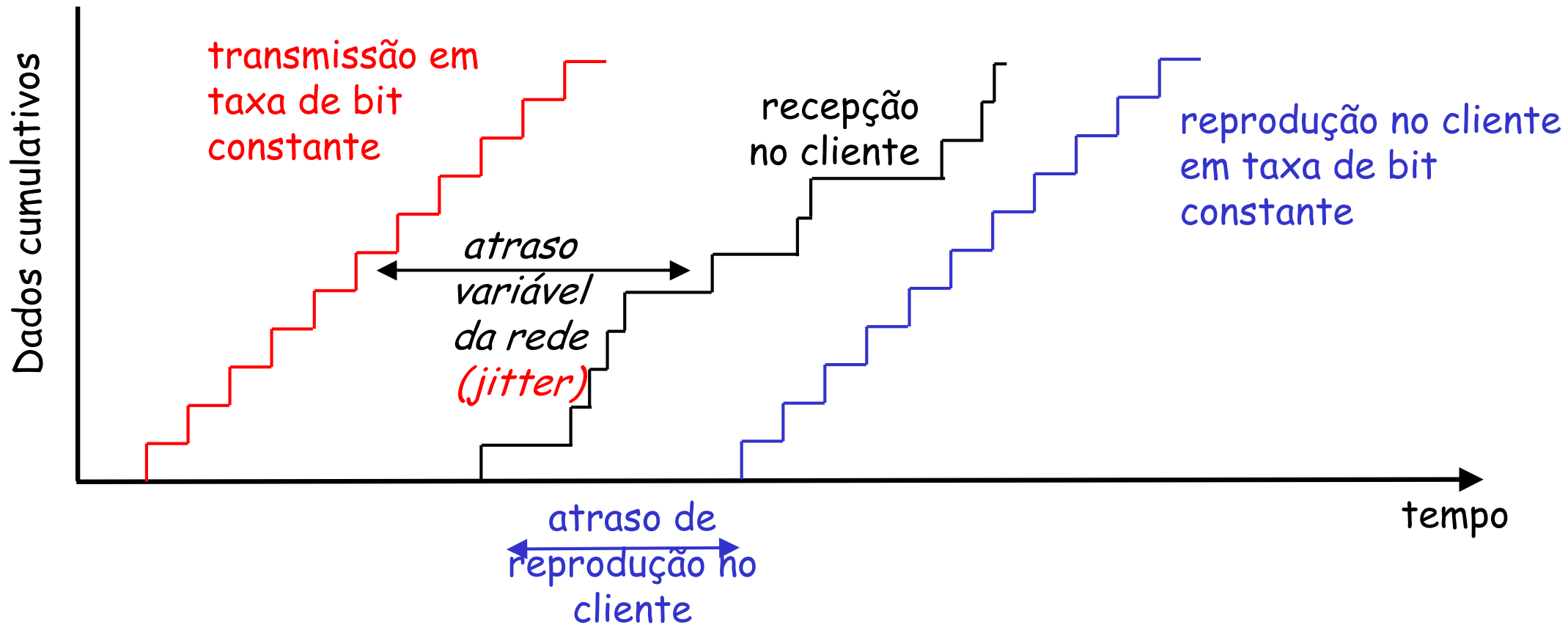


PEARSON

Addison  
Wesley

## 7

## Jitter



- Considere os atrasos fim-a-fim de dois pacotes consecutivos: diferença pode ser mais ou menos do que 20 msec

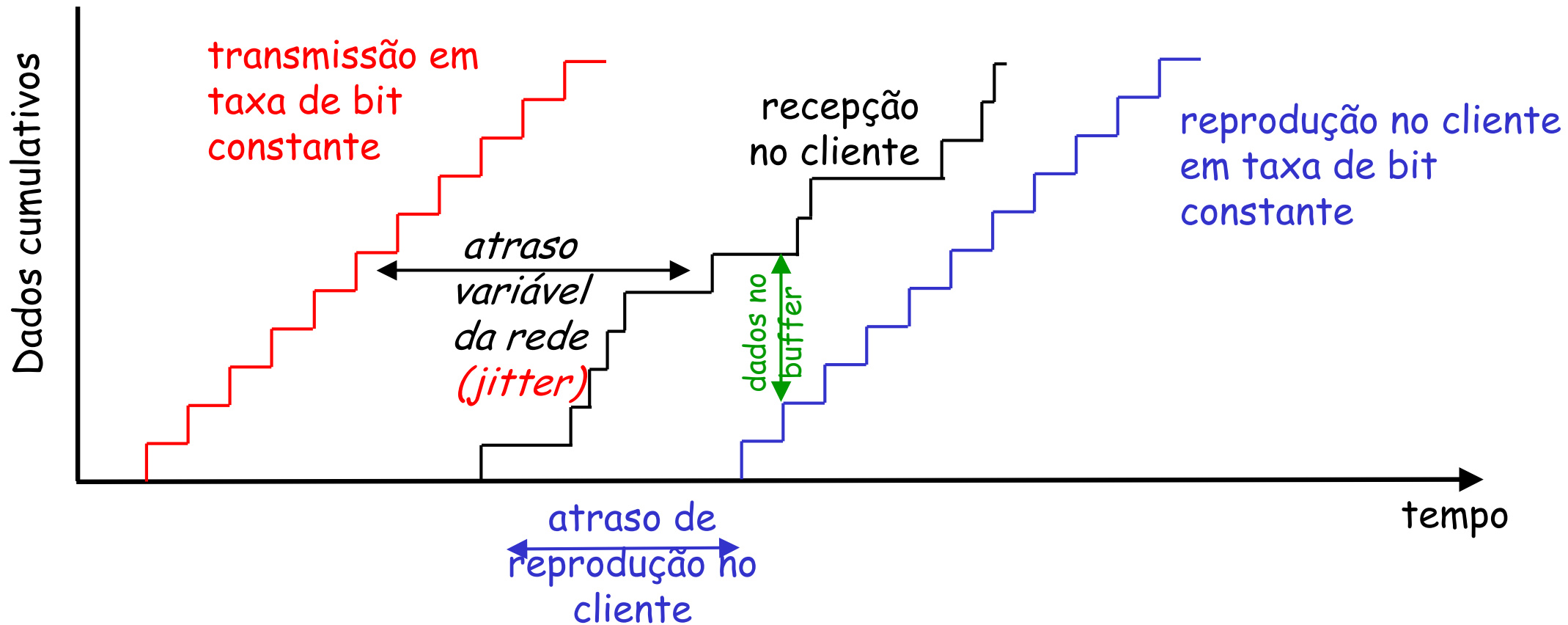


PEARSON

Addison  
Wesley

## 7

## Jitter



- Considere os atrasos fim-a-fim de dois pacotes consecutivos: diferença pode ser mais ou menos do que 20 msec



PEARSON

Addison  
Wesley

# 7

## Telefonia Internet: atraso fixo de reprodução

- Receptor tenta reproduzir cada bloco exatamente  $q$  msecs após o bloco ser gerado.
  - Bloco possui marca de tempo  $t$ : bloco de reprodução em  $t + q$ .
  - Bloco chega após  $t + q$ : dado chega muito tarde para ser executado, “perda” do dado
- Compromisso de  $q$ :
  - $Q$  grande: menor perda de pacote
  - $Q$  pequeno: melhor experiência interativa

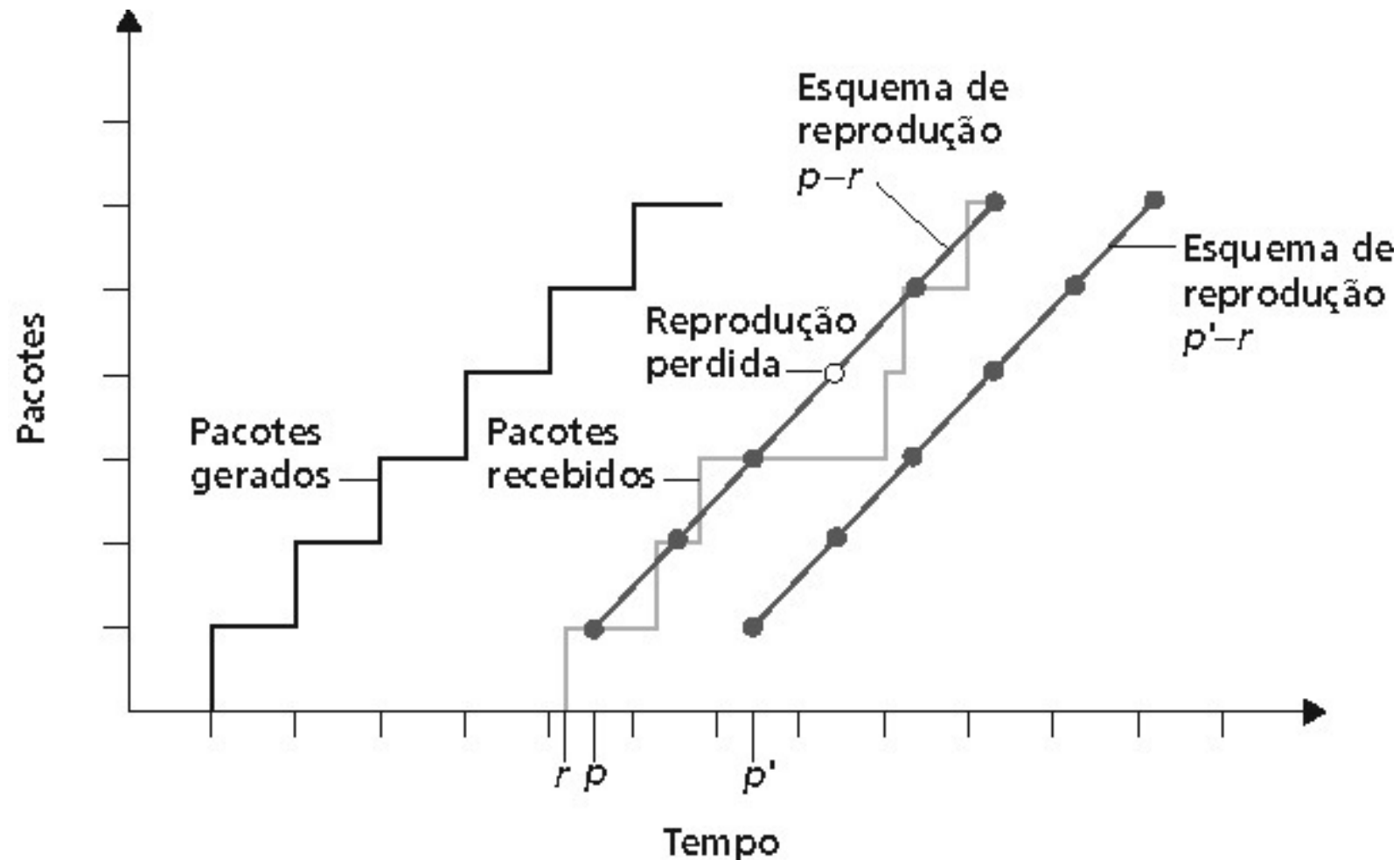


# 7 Atraso fixo de reprodução

Transmissor gera pacotes a cada 20 ms durante os intervalos de atividade.  
Primeiro pacote é recebido no instante  $r$ .

Primeira programação de reprodução: começa em  $p$ .

Segunda programação de reprodução: começa em  $p'$ .





# 7 Atraso de reprodução adaptativo (1)

- Estima o atraso da rede e ajusta o atraso de reprodução no início de cada intervalo de atividade
- Intervalos de silêncio são aumentados e diminuídos
- Blocos ainda são gerados a cada 20 ms nos intervalos de atividade

$t_1$  = marca de tempo do  $i$ -ésimo pacote

$r_1$  = instante no qual o pacote  $i$  é recebido pelo receptor

$p_1$  = instante no qual o pacote  $i$  é reproduzido no receptor

$r_1 - t_1$  = atraso da rede para o  $i$ -ésimo pacote

$d_1$  = estimativa do atraso na rede após receber o  $i$ -ésimo pacote

Estimativa dinâmica do atraso médio no receptor:

$$d_i = (1 - u)d_{i-1} + u(r_i - t_i)$$

Em que  $u$  é uma constante fixa (ex.,  $u = 0,01$ ).

# 7 Atraso de reprodução adaptativo (2)

É também usual estimar a variância média do atraso,  $v_i$ :

$$v_i = (1 - u)v_{i-1} + u |r_i - t_i - d_i|$$

As estimativas de  $d_i$  e  $v_i$  são calculadas para cada pacote recebido, embora elas sejam usadas apenas no início de um intervalo de atividade.

Para o primeiro pacotes de um intervalo de atividade, o instante de reprodução é:

$$p_i = t_i + d_i + Kv_i$$

Em que  $K$  é uma constante positiva constante.

Os pacotes restantes no intervalo de atividade são reproduzidos periodicamente

# 7 Atraso de reprodução adaptativo (3)

Como saber se um pacote é o primeiro de um intervalo de atividade:

- Se nunca houvesse perdas, o receptor poderia simplesmente olhar nas marcas de tempo sucessivas.
  - Se a diferença de marcas de tempo sucessivas for maior que 20 ms, então temos o início de um intervalo de atividade.
- Mas porque as perdas são possíveis, o receptor deve olhar tanto as marcas de tempo como os números de sequência dos pacotes.
  - Se a diferença de marcas de tempo sucessivas é maior que 20 ms e não há pulos nos números de sequência, então tem-se o início de um intervalo de atividade.

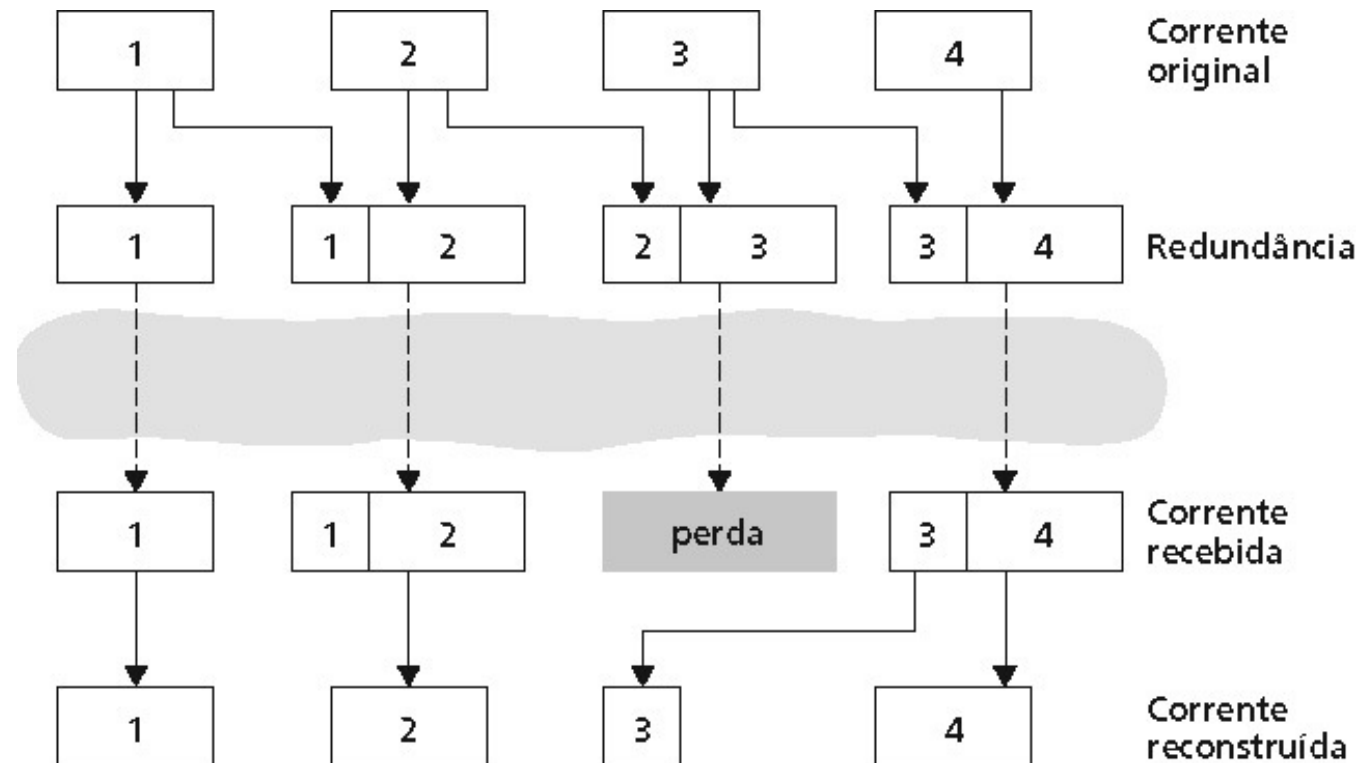
## Correção de erro de envio (FEC): esquema simples

- Para cada grupo de  $n$  blocos, cria um bloco redundante realizando uma operação OU exclusivo entre os  $n$  blocos originais
- Envia os  $n + 1$  blocos aumentando a banda passante por um fator de  $1/n$
- Pode reconstruir os  $n$  blocos originais se houver no máximo um bloco perdido nos  $n+1$  blocos enviados
- Atraso de reprodução precisa ser definido para receber todos os  $n + 1$  pacotes
- Compromisso:
  - Aumentar  $n$ , menor desperdício de banda
  - Aumentar  $n$ , maior atraso de reprodução
  - Aumentar  $n$ , maior a probabilidade que dois ou mais blocos sejam perdidos

## 7

# Recuperação de perdas de pacotes (2)

- 2º esquema FEC
- Enviar um fluxo de menor qualidade como “carona”
- Enviar fluxo de áudio de menor resolução como a informação redundante
- Por exemplo, um fluxo PCM nominal a 64 kbps e um fluxo GSM redundante a 13 kbps

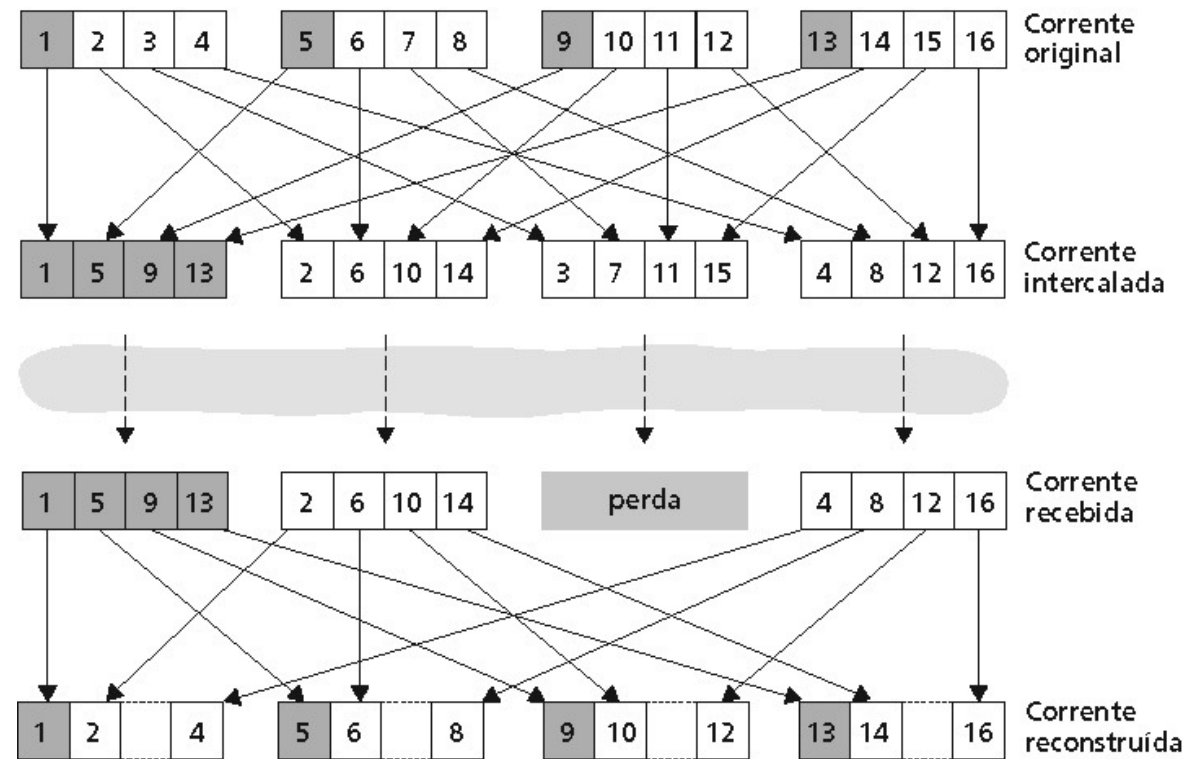


- Sempre que ocorre perda não consecutiva, o receptor pode esconder a perda
- Pode também anexar os blocos  $(n - 1)$  e  $(n - 2)$  do fluxo de baixa qualidade



# 7

## Recuperação de perdas de pacotes (3)



### Interpolação

- Blocos são quebrados em unidades menores
- Por exemplo, 4 blocos de 5 ms cada
- Pacote agora contém unidades menores de diferentes blocos
- Se o pacote for perdido, ainda resta mais de cada bloco
- Não há cabeçalho de redundância
- Mas adiciona ao atraso de reprodução

- Usa UDP para evitar o controle de congestionamento do TCP (atrasos) para tráfego sensível ao tempo
- Atraso de reprodução adaptivo do lado cliente: para compensar atrasos
- Lado servidor ajusta a largura de banda do fluxo à largura de banda disponível no caminho cliente-ao-servidor
  - Escolhe entre taxas de fluxo pré-codificadas
  - Taxa codificação do servidor dinâmica
- Recuperação de erros (no topo do UDP)
  - FEC, interpolação
  - Retransmissões, permissão de tempo
  - Esconde erros: repete dados próximos



# 7 Redes multimídia

- 7.1 Aplicações de Rede Multimídia
- 7.2 Transmissão em fluxo contínuo de áudio e video armazenados
- 7.3 Multimídia em tempo real: estudo da Telefonia Internet
- 7.4 Protocolos para Aplicações Interativas em tempo real
  - RTP, RTCP, SIP
- 7.5 Distribuição de Multimídia: redes de distribuição de conteúdo
- 7.6 Além do Melhor Esforço
- 7.7 Agendamento e Mecanismos de Policiamento
- 7.8 Serviços Integrados e Serviços Diferenciados
- 7.9 RSVP



# 7 Protocolo de tempo real (RTP)

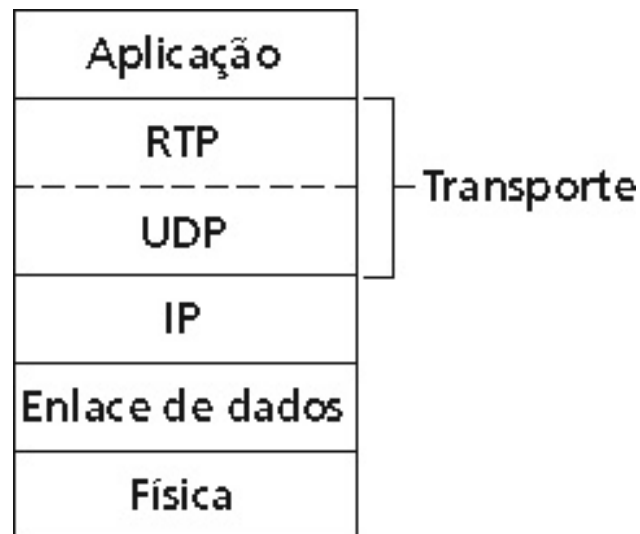
- RTP especifica uma estrutura de pacotes que transportam dados de áudio e vídeo
- RFC 1889
  - Pacote RTP oferece
    - Identificação do tipo de carga
    - Numeração da seqüência de pacotes
    - Marcas de tempo
- RTP roda nos sistemas terminais.
- Os pacotes RTP são encapsulados em segmentos UDP
- Interoperabilidade: se duas aplicações de telefonia IP usam RTP, então elas podem ser capazes de trabalhar juntas

# 7

## RTP roda em cima do UDP

As bibliotecas do RTP fornecem uma interface de camada de transporte que estendem o UDP:

- Número de portas, endereços IP
- Identificação do tipo de carga
- Numeração da sequência de pacotes
- Marcas de tempo



- Considere enviar 64 kbps de voz codificada em PCM sobre RTP
- Aplicação reúne dados codificados em blocos, por exemplo, a cada 20 ms = 160 bytes por bloco
- O bloco de áudio, junto com o cabeçalho RTP forma o pacote RTP, que é encapsulado num segmento UDP
- O cabeçalho RTP indica o tipo de codificação de áudio em cada pacote
  - Os transmissores podem mudar a codificação durante a conferência
- O cabeçalho RTP também contém os números de seqüência e marcas de tempo



# 7 RTP e QoS

- RTP **não** fornece nenhum mecanismo para assegurar a entrega dos pacotes e dados no tempo correto, nem fornece outras garantias de qualidade de serviço
- O encapsulamento RTP é visto apenas nos sistemas finais - ele não é percebido pelos roteadores intermediários
  - Roteadores fornecem o serviço de melhor esforço tradicional da Internet. Eles não fazem nenhum esforço especial para assegurar que os pacotes RTP cheguem no destino no momento correto

Tipo de carga útil	Número de sequência	Marca de tempo	Identificador de sincronização da fonte	Campos variados
--------------------	---------------------	----------------	---	-----------------

## Cabeçalho RTP

**Tipo de Carga (7 bits):** Usado para indicar o tipo de codificação que está sendo usado no momento. Se um transmissor muda o tipo de codificação durante uma conferência, o transmissor informa o receptor através deste campo de tipo de carga.

- Tipo de carga 0: PCM mu-law, 64 kbps
- Tipo de carga 3, GSM, 13 kbps
- Tipo de carga 7, LPC, 2.4 kbps
- Tipo de carga 26, Motion JPEG
- Tipo de carga 31. H.261
- Tipo de carga 33, MPEG2 vídeo

**Número de sequência (16 bits):** O número de sequência é incrementado de um a cada pacote RTP enviado; pode ser usado para detectar perdas de pacotes e para recuperar a sequência de pacotes.

- **Timestamp field (32 bytes long).** Reflete o instante de amostragem do primeiro byte no pacote de dados RTP.
  - Para áudio o relógio de marca de tempo incrementa de um a cada intervalo de amostragem (por exemplo, cada 125 us para uma taxa de amostragem de 8 KHz)
  - Se a aplicação de áudio gera blocos contendo 160 amostras codificadas, então a marca de tempo do RTP aumenta de 160 para cada pacote RTP quando a fonte está ativa. O relógio de marca de tempo continua a aumentar numa taxa constante mesmo quando a fonte está inativa
- **Campo SSRC (32 bits).** Identifica a fonte do fluxo RTP. Cada fluxo numa sessão RTP deve ter um SSRC distinto



# 7 Tarefa de programação RTSP/RTP

- Construir um servidor que encapsula quadros de vídeo armazenado dentro de pacotes RTP
  - Pegue o quadro de vídeo, adicione cabeçalhos RTP, crie segmentos UDP, envie os segmentos para o socket UDP
  - Inclua números de seqüência e marcas de tempo
  - O cliente RTP é fornecido para você
- Escreva também o lado cliente do RTSP
  - Edite comandos de reproduzir e parar
  - O servidor RTSP é fornecido para você

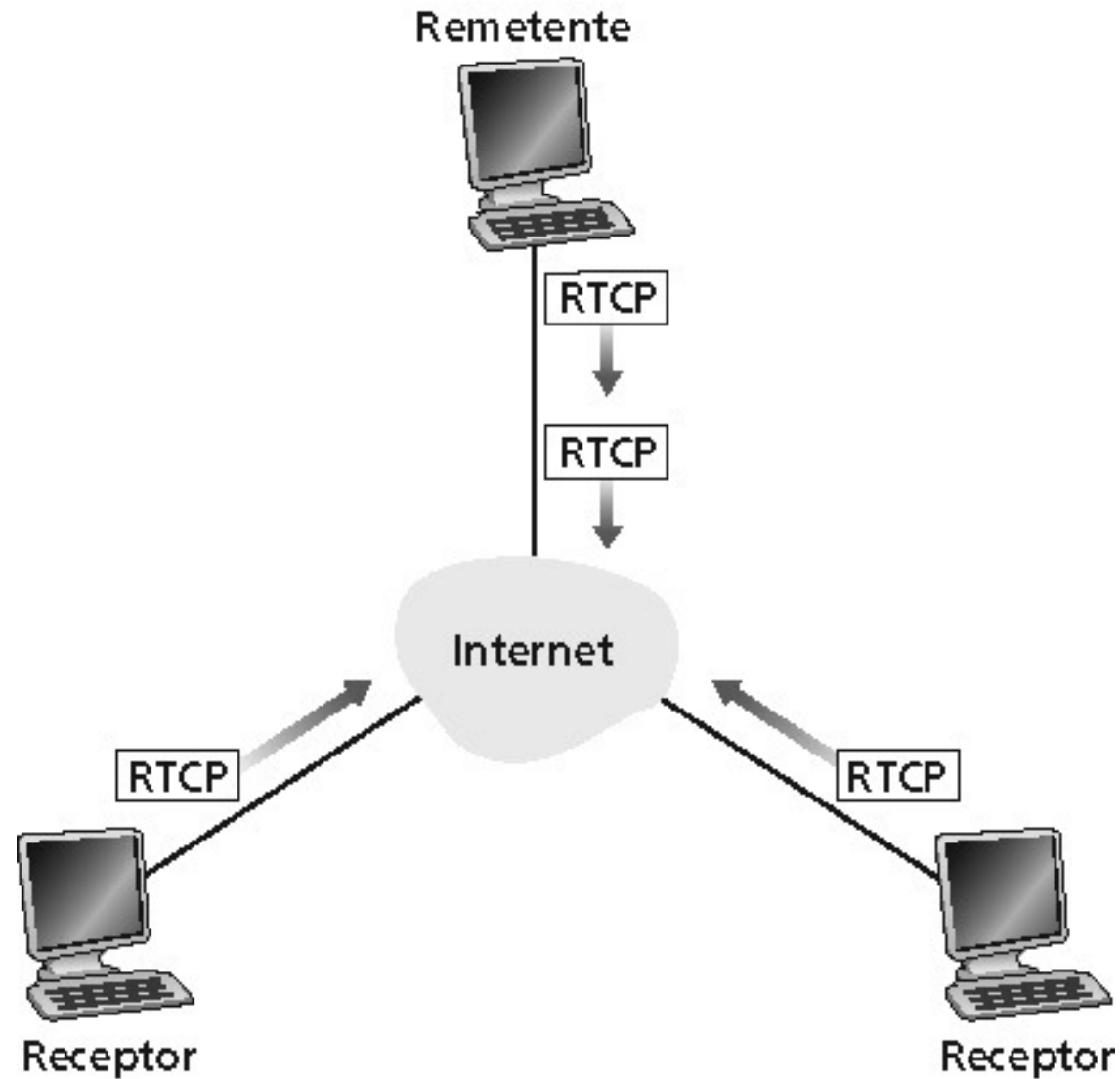
# 7 Real-Time Control Protocol (RTCP)

- Trabalha em conjunto com o RTP
- Cada participante de uma sessão RTP transmite periodicamente pacotes de controle RTCP para todos os outros participantes
- Cada pacote RTCP contém relatórios do transmissor e/ou do receptor
  - Estatísticas de relatório são úteis para a aplicação
- As estatísticas incluem o número de pacotes enviados, número de pacotes perdidos, variação de atraso entre chegadas etc.
- Esta informação de realimentação para a aplicação pode ser usada para controle do desempenho e para fins de diagnóstico
  - O transmissor pode mudar suas transmissões com base nestas informações de realimentação



# 7 RTCP (Cont.)

- Para uma sessão RTP existe tipicamente um único endereço de multicast; todos os pacotes RTP e RTCP pertencentes à sessão usam este endereço de multicast.
- Os pacotes RTP e RTCP são distintos um dos outros pelo uso de números de portas diferentes.
- Para limitar o tráfego, cada participante reduz seu tráfego RTCP quando o número de participantes da conferência aumenta.



## Pacotes de relatório do receptor:

- Fração de pacotes perdidos, último número de sequência, variância média do atraso entre chegadas

## Pacotes de relatório do transmissor:

- SSRC do fluxo RTP, o tempo corrente, o número de pacotes enviados e o número de bytes enviados

## Pacotes de descrição da fonte:

- Endereço de e-mail do transmissor, o nome do transmissor, o SSRC do fluxo RTP associado
- Fornecem um mapeamento entre o SSRC e o nome do usuário ou do hospedeiro



# 7 Sincronização de fluxos

- RTCP pode ser usado para sincronizar diferentes fluxos de mídia numa sessão RTP
- Considere uma aplicação de videoconferência para a qual cada transmissor gera um fluxo RTP para áudio e um para vídeo
- As marcas de tempo nesses pacotes são vinculadas aos relógios de amostragem de vídeo e de áudio, mas não são vinculadas a um relógio de tempo real (isto é, a um relógio de parede)
- Cada pacote relatório do transmissor RTCP contém (para o último pacote gerado no fluxo RTP associado):
  - Marca de tempo do pacote RTP
  - Instante de tempo real no qual o pacote foi criado
- Receptores podem usar esta associação para sincronizar a reprodução de áudio e de vídeo



- O RTCP procura limitar seu tráfego a 5% da banda passante da sessão

## Exemplo

- Suponha que existe um transmissor enviando vídeo com uma taxa de 2 Mbps. Então o RTCP procura limitar seu tráfego a 100 kbps
- RTCP dá 75% dessa taxa para os receptores; 25% do restante para o transmissor
- Os 75 kbps dedicados aos receptores são divididos de forma igual entre os receptores :
  - Com  $R$  receptores, cada receptor consegue enviar tráfego RTCP a uma taxa de  $75/R$  kbps
- Transmissor envia tráfego RTCP a uma taxa de 25 kbps
- Um participante (um transmissor ou receptor) determina o período de transmissão de pacotes RTCP dinamicamente calculando o tamanho médio do pacote (durante toda a sessão) e dividindo o tamanho médio do pacote RTCP pela sua taxa alocada



# 7 SIP

- Session Initiation Protocol
- Desenvolvido pelo IETF

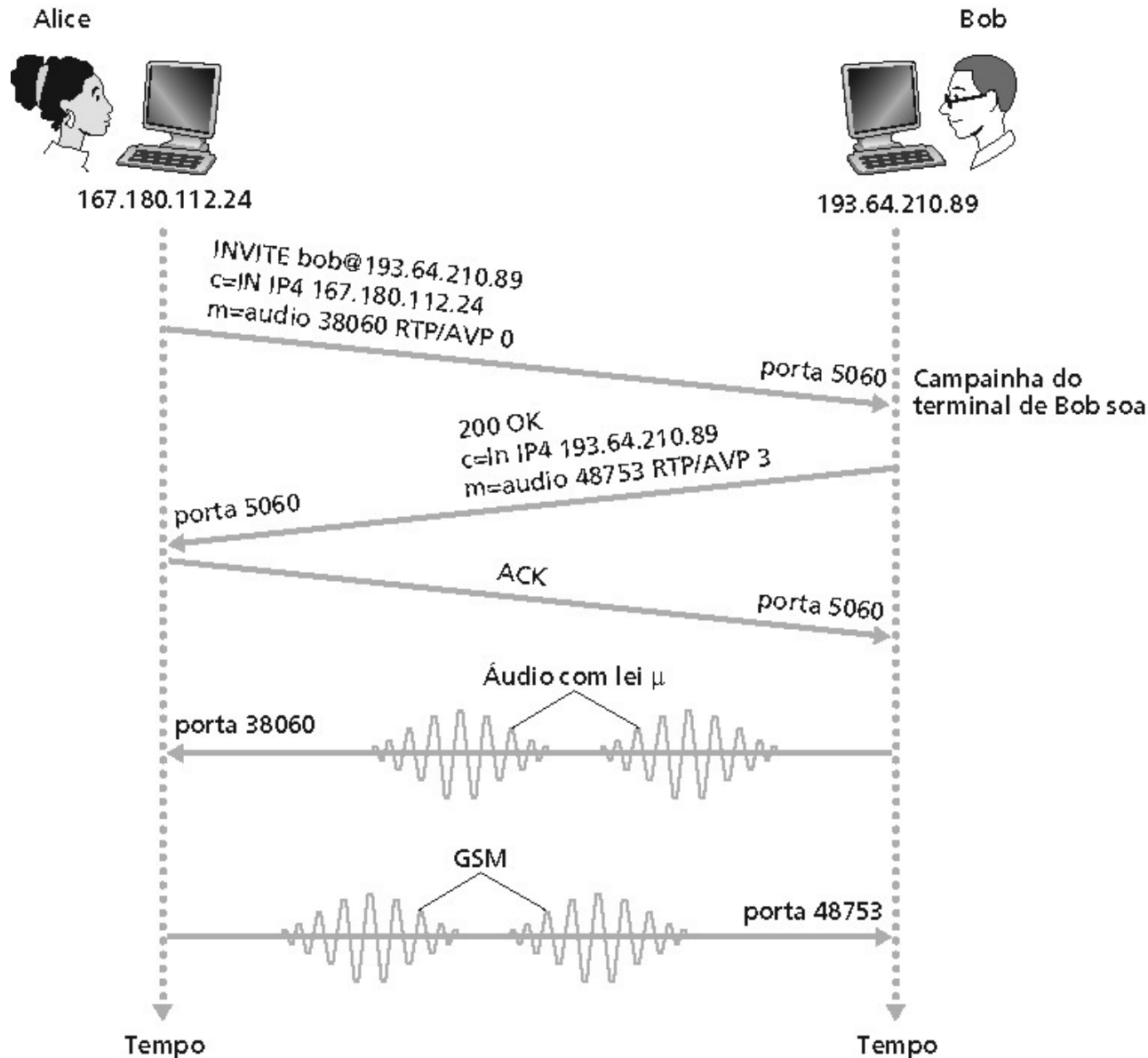
## Visão de longo prazo do SIP

- Todas chamadas telefônicas e chamadas de videoconferência ocorrem sobre a Internet
- Pessoas são identificadas por nomes ou endereços de e-mail, em vez de números telefônicos.
- Você pode alcançar o usuário chamado, não importa onde ele esteja, não importa o dispositivo IP que ele esteja usando atualmente.

- Estabelecendo uma chamada
  - Provê mecanismos para o chamador deixar o usuário chamado saber que ele deseja estabelecer uma chamada
  - Provê mecanismos de modo que o chamador e o chamado possam concordar com o tipo de mídia e codificação.
  - Provê mecanismos para terminar a chamada.
- Determina o endereço IP do usuário chamado.
  - Mapeia o identificador mnemônico para o endereço IP atual
- Gerenciamento de chamada
  - Adiciona novos fluxos de mídia durante a chamada
  - Troca a codificação durante a chamada
  - Convida outros
  - Transfere e retém chamadas



# 7 Estabelecendo uma chamada para um endereço IP conhecido



- A mensagem *INVITE* do SIP de Alice indica seu número de porta e endereço IP. Indica a codificação que Alice prefere receber (PCM lei-m)
- A mensagem *200 OK* de Bob indica seu número de porta, endereço IP e codificação preferida (GSM)
- Mensagens SIP podem ser enviadas sobre TCP ou UDP; aqui são enviadas sobre RTP/UDP.
- O número de porta padrão do SIP é 5060.

# 7

## Estabelecendo uma chamada (mais)

- Negociação do codec:
  - Suponha que Bob não tenha o codificador do PCM para lei m
  - Bob responderá então com *606 Not Acceptable Reply* e listará os codificadores que ele pode usar
  - Alice pode então enviar uma nova mensagem *INVITE*, anunciando um codificador apropriado
- Rejeitando a chamadal
  - Bob pode rejeitar com respostas “ocupado,” “ausente,” “pagamento exigido,” “proibido”
- A mídia pode ser enviada sobre RTP ou algum outro protocolo



# 7

## Exemplo de mensagem SIP

```
INVITE sip:bob@domain.com SIP/2.0
Via: SIP/2.0/UDP 167.180.112.24
From: sip:alice@hereway.com
To: sip:bob@domain.com
Call-ID: a2e3a@pigeon.hereway.com
Content-Type: application/sdp
Content-Length: 885
```

```
c=IN IP4 167.180.112.24
m=audio 38060 RTP/AVP 0
```

- Aqui não sabemos o endereço IP de Bob. Servidores SIP intermediários serão necessários
- Alice envia e recebe mensagens SIP usando o número 5060 de porta padrão do SIP
- Alice especifica através de cabeçalho que o cliente SIP envia e recebe mensagens SIP sobre UDP

### Notas:

- Sintaxe de mensagem HTTP
- sdp = protocolo de descrição de sessão
- ID de chamada (*Call-ID*) é único para cada chamada.

- O chamador quer chamar o usuário de destino, mas tem somente o nome do usuário ou o endereço de e-mail
- Precisa obter o endereço IP do hospedeiro atual do usuário chamado:
  - Usuário move-se ao redor
  - Protocolo DHCP
  - Usuário possui diferentes dispositivos IP (PC, PDA, dispositivo de carro)
- Resultado pode ser baseado em:
  - Hora do dia (trabalho, casa)
  - Usuário chamado (não quer ser chamada pelo chefe em casa)
  - Status do chamado (chamadas enviadas para correio-de-voz quando o chamado já está falando com alguém)

## Serviços fornecidos pelos servidores SIP:

- Servidor de registro SIP
- Servidor proxy SIP



# 7

## Registro SIP

- Quando Bob inicia o cliente SIP, o cliente envia a mensagem *SIP REGISTER* ao servidor de registro de Bob (função similar necessária para mensagens instantâneas)

### Mensagem de registro:

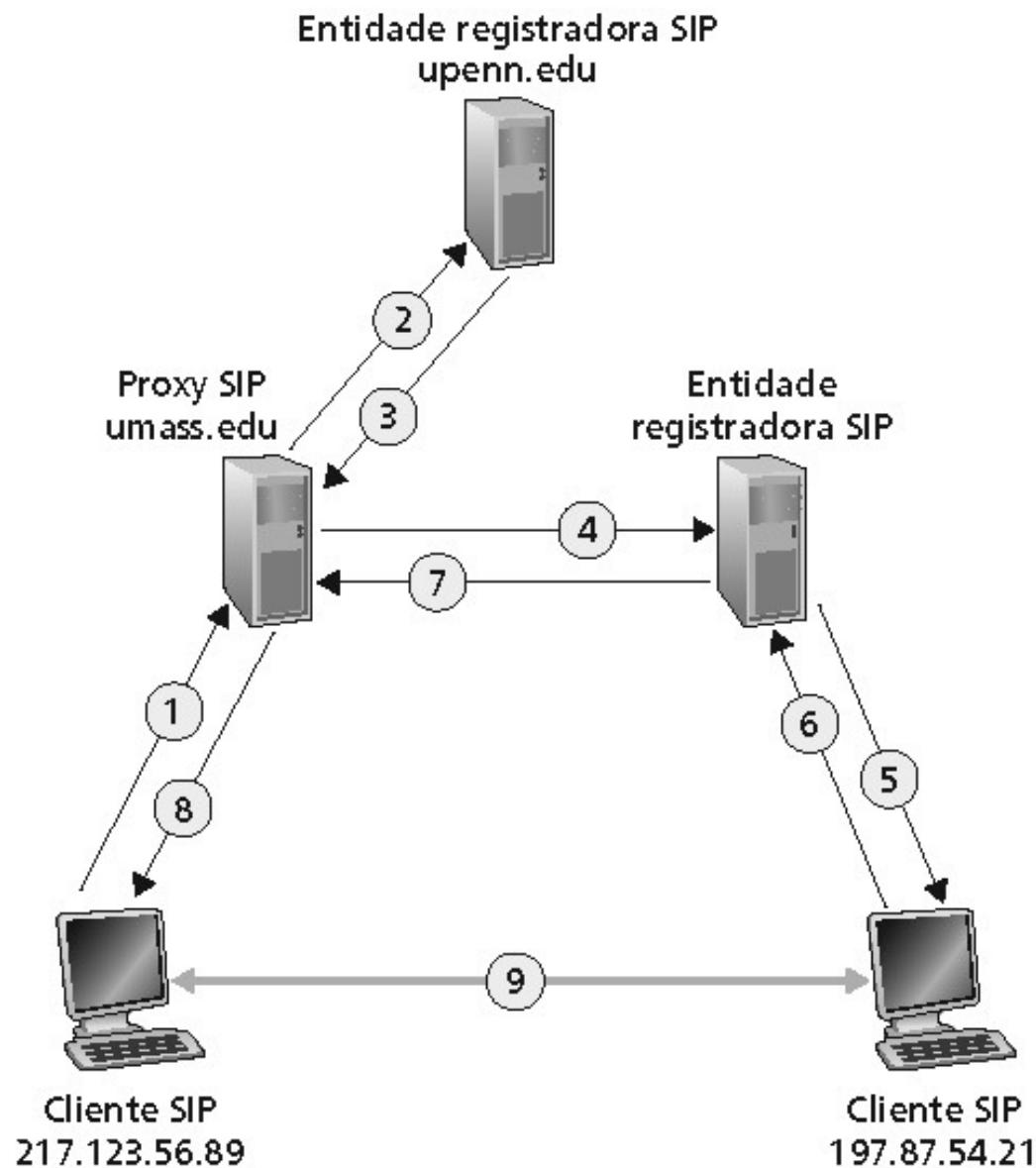
```
REGISTER sip:domain.com SIP/2.0  
Via: SIP/2.0/UDP 193.64.210.89  
From: sip:bob@domain.com  
To: sip:bob@domain.com  
Expires: 3600
```



- Alice envia a mensagem “invite” para o seu servidor proxy
  - Contém endereço sip:bob@domain.com
- Proxy responsável por rotear mensagens SIP para o usuário chamado
  - Possivelmente através de múltiplos proxies.
- O usuário chamado envia a resposta de volta através do mesmo conjunto de proxies.
- Proxy retorna a mensagem de resposta SIP para Alice
  - Contém o endereço IP de Bob
- Nota: proxy é análogo ao servidor DNS local

# 7

## Exemplo



Usuário chamado `jim@umass.edu` que estabelece uma chamada para `keith@upenn.edu`

(1) Jim envia mensagem *INVITE* para o proxy SIP umass SIP.  
 (2) Proxy encaminha a requisição para o servidor de registro upenn.  
 (3) servidor upenn retorna resposta, indicando que ele deveria tentar `keith@eurecom.fr`  
 (4) proxy umass envia *INVITE* para registro eurecom. (5) registro eurecom encaminha *INVITE* para 197.87.54.21, que está rodando o cliente SIP de keith.  
 (6-8) resposta SIP enviada de volta (9) mídia enviada diretamente entre clientes.

**Nota:** também há mensagens *ACK* SIP, que não são mostradas.

- H.323 é outro protocolo de sinalização para tempo real, interativo
- H.323 é um conjunto de protocolos completos e verticalmente integrados para conferência multimídia: sinalização, registro, controle de admissão, transporte e codecs.
- SIP é um componente monolítico. Funciona com RTP, mas não o obriga. Pode ser combinado com outros protocolos e serviços.
- H.323 foi desenvolvido pela ITU (telefonia).
- SIP foi desenvolvido pela IETF: Muitos de seus conceitos vêm do HTTP. O SIP é parecido com a Web, visto que H.323 é parecido com a telefonia.
- SIP usa o princípio de KISS (*keep it simple stupid*).

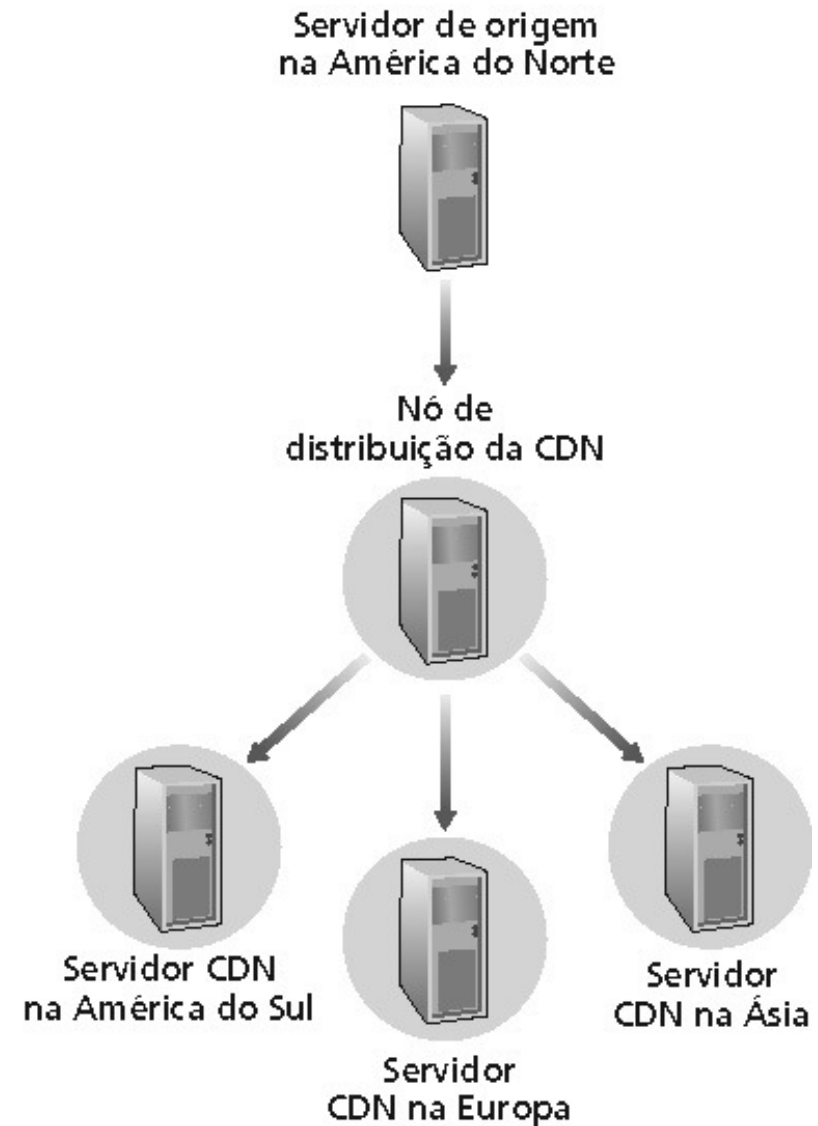


# 7 Redes multimídia

- 7.1 Aplicações de Rede Multimídia
- 7.2 Transmissão em fluxo contínuo de áudio e video armazenados
- 7.3 Multimídia em tempo real: estudo da Telefonia Internet
- 7.4 Protocolos para Aplicações Interativas em tempo real RTP, RTCP, SIP
- 7.5 Distribuição de Multimídia: redes de distribuição de conteúdo
- 7.6 Além do Melhor Esforço
- 7.7 Agendamento e Mecanismos de Policiamento
- 7.8 Serviços Integrados e Serviços Diferenciados
- 7.9 RSVP

## Réplica de conteúdo

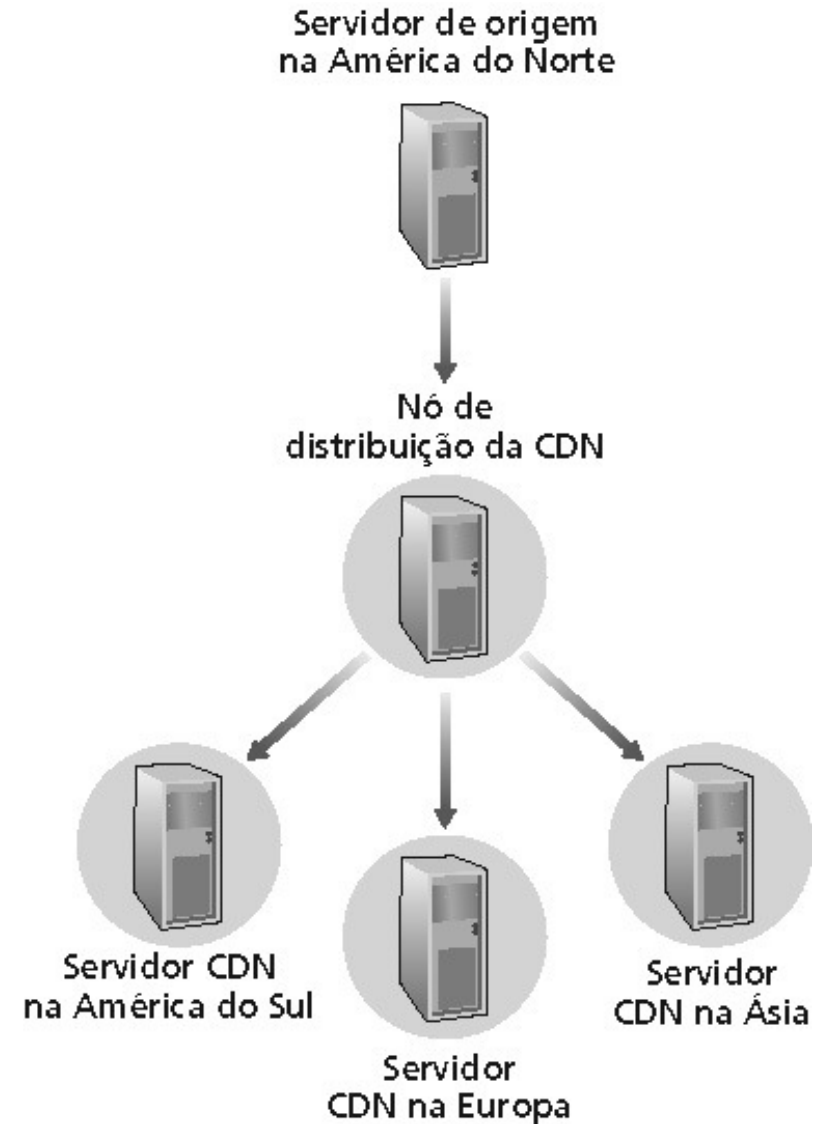
- Desafio para transmitir fluxo de arquivos grandes (ex., vídeo) de um único servidor de origem em tempo real
- Solução: replica o conteúdo em centenas de servidores por toda a Internet
  - Conteúdo é descarregado nos servidores CDN antes do tempo de uso
  - Colocar o conteúdo "próximo" ao usuário evita prejuízos (perda, atraso) de se enviar o conteúdo por longos caminhos
  - Servidor CDN tipicamente na rede de borda/acesso





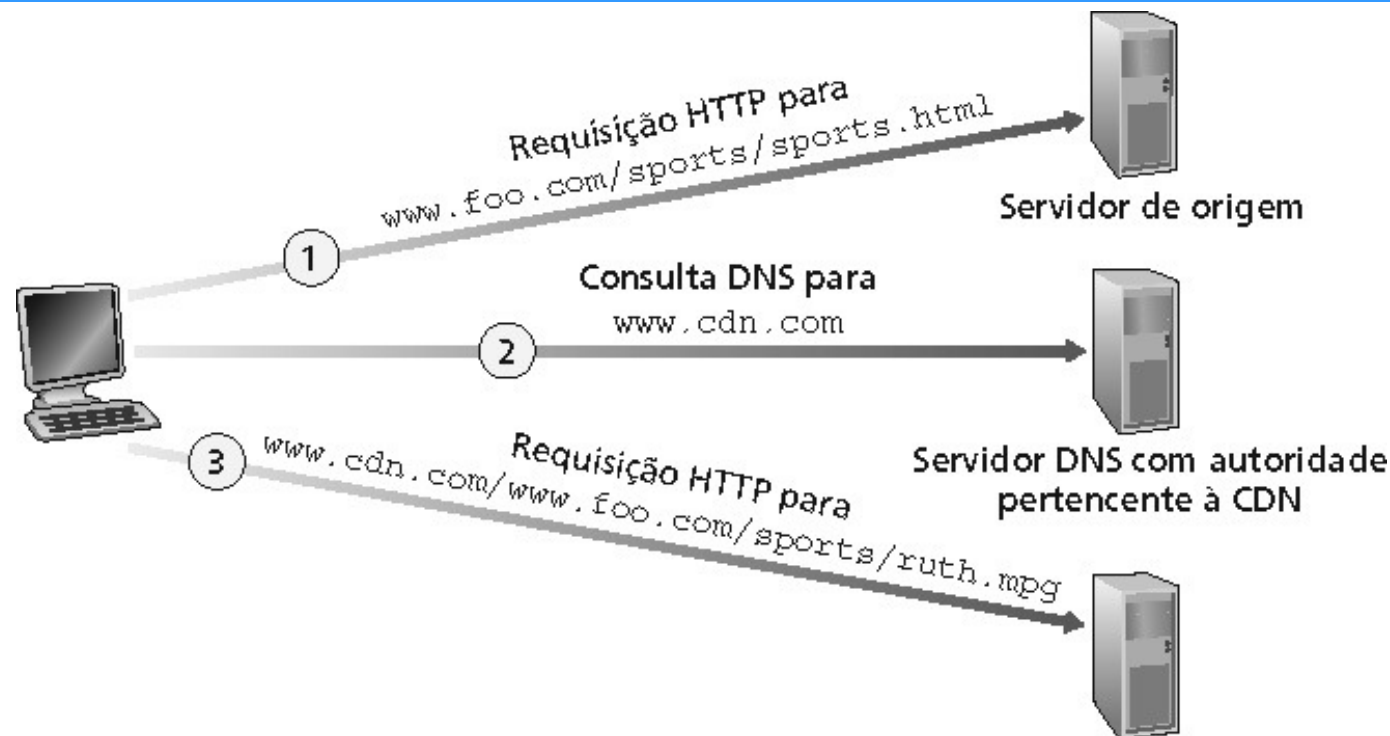
## Réplica de conteúdo

- Cliente CDN (ex., Akamai) é o provedor de conteúdo (ex., CNN)
- CDN replica o conteúdo dos clientes em servidores CDN. Quando o provedor atualiza o conteúdo, a CDN atualiza os servidores



## 7

## CDN: exemplo

**Servidor de origem (www.foo.com)**

- Distribui HTML
- Substitui:  
`http://www.foo.com/sports.ruth.gif`  
por  
`http://www.cdn.com/www.foo.com/sports/ruth.gif`

**Companhia CDN (cdn.com)**

- Distribui arquivos gif
- Usa seu servidor DNS autoritário para rotear requisições redirecionadas



# 7 Mais sobre CDNs

## Requisições de roteador

- CDN cria um “mapa”, indicando as distâncias dos ISPs aos nós CDN
- Quando a consulta chega ao servidor DNS autoritário:
  - Servidor determina ISP de onde se originou a consulta
  - Utiliza o “mapa” para determinar o melhor servidor CDN
- Nós CDN criam a rede de sobreposição da camada de aplicação

# 7 Redes multimídia

- 7.1 Aplicações de Rede Multimídia
- 7.2 Transmissão em fluxo contínuo de áudio e video armazenados
- 7.3 Multimídia em tempo real: estudo da Telefonia Internet
- 7.4 Protocolos para Aplicações Interativas em tempo real RTP, RTCP, SIP
- 7.5 Distribuição de Multimídia: redes de distribuição de conteúdo
- 7.6 Além do Melhor Esforço
- 7.7 Agendamento e Mecanismos de Policiamento
- 7.8 Serviços Integrados e Serviços Diferenciados
- 7.9 RSVP

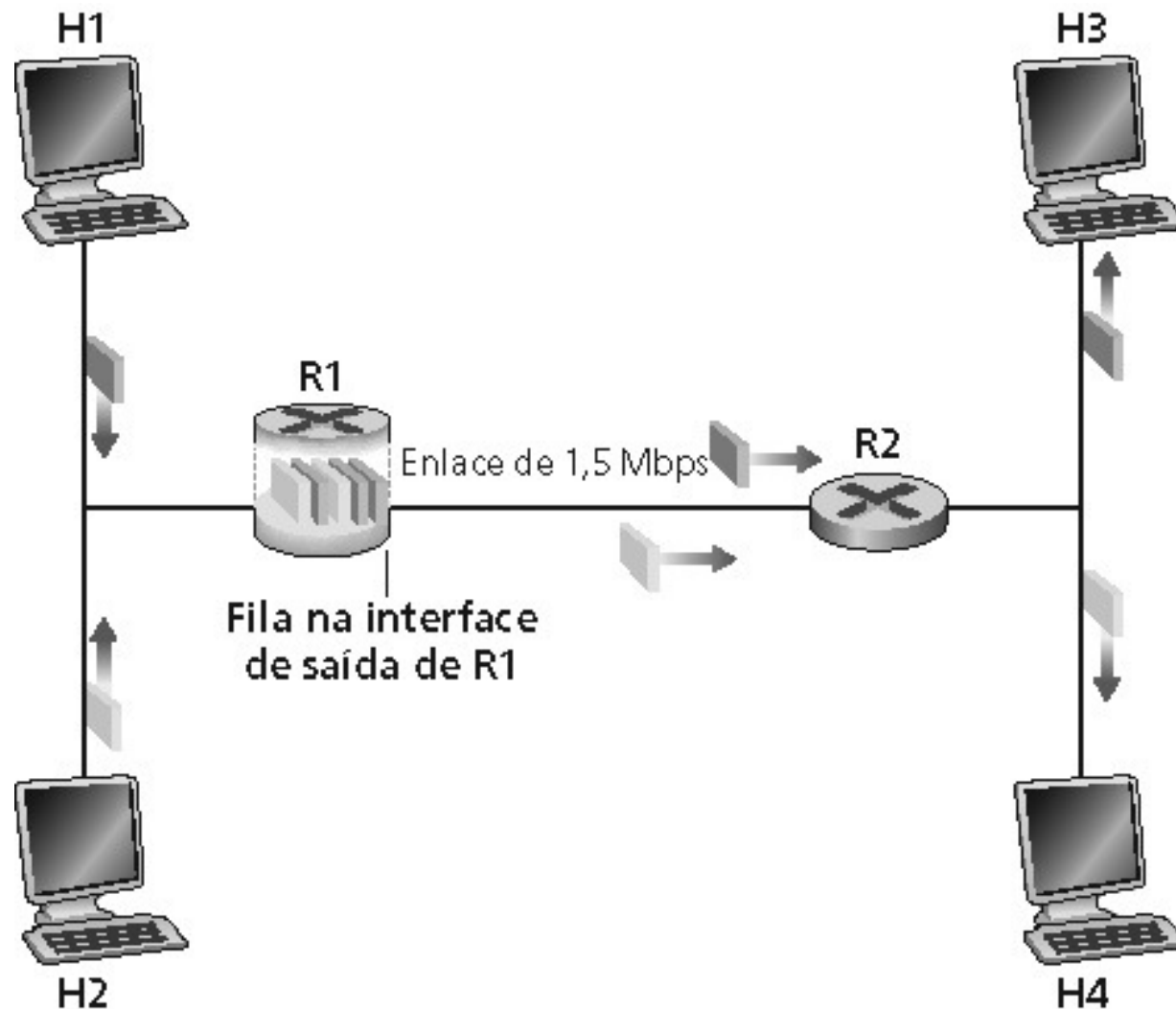
# 7 Oferecendo QoS em Redes IP

**Até agora:** “fazer o melhor do melhor esforço”

**Futuro:** Internet da próxima geração com garantias de QoS

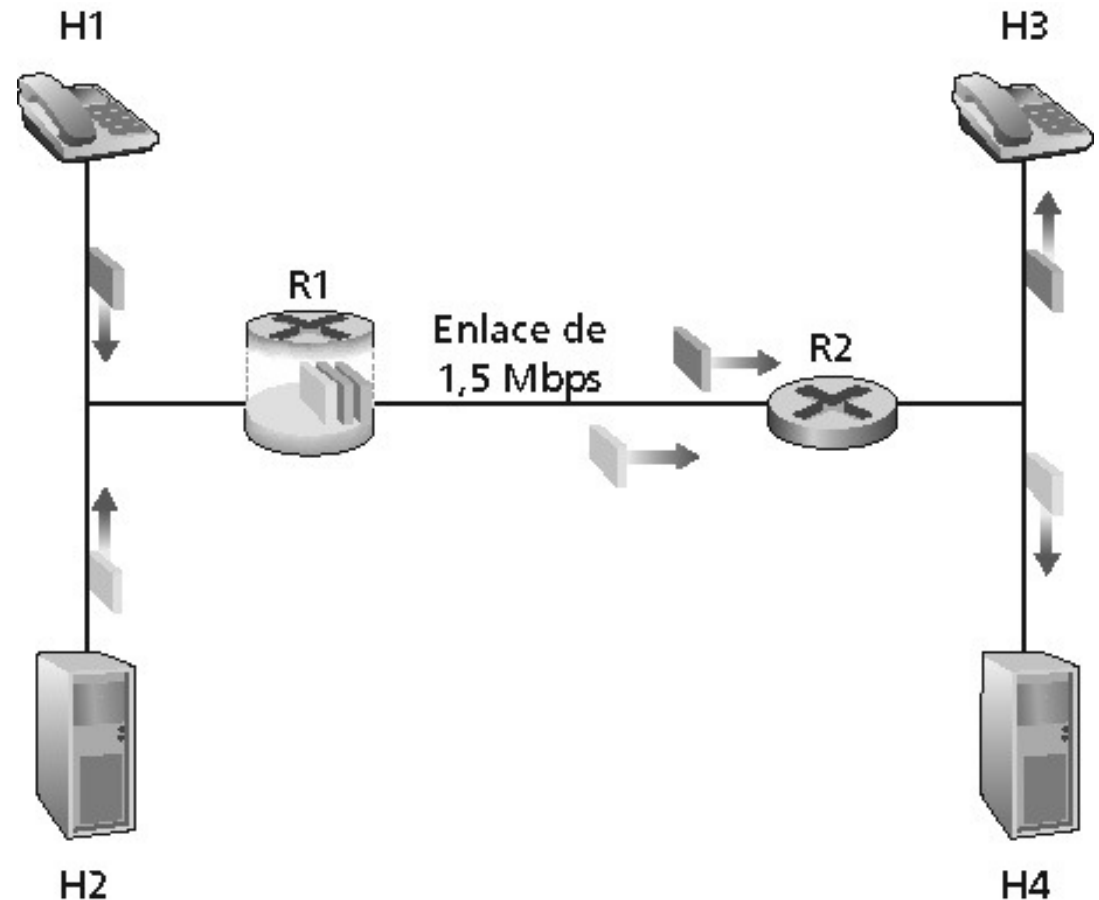
- **RSVP:** sinalização para reserva de recursos
- **Serviços diferenciados:** garantias diferenciais
- **Serviços integrados:** garantias rígidas

Modelo simples para estudos de compartilhamento e congestionamento:



# 7 Princípios para garantias de QoS

- Exemplo: 1 Mbps para telefonia IP, FTP compartilha um enlace de 1,5 Mbps
- Rajadas de FTP podem congestionar o roteador, causando perda de áudio
- Quer dar prioridade ao áudio sobre FTP



## Princípio 1

Marcação dos pacotes é necessária para o roteador distinguir entre diferentes classes; assim como novas regras de roteamento para tratar os pacotes apropriadamente.

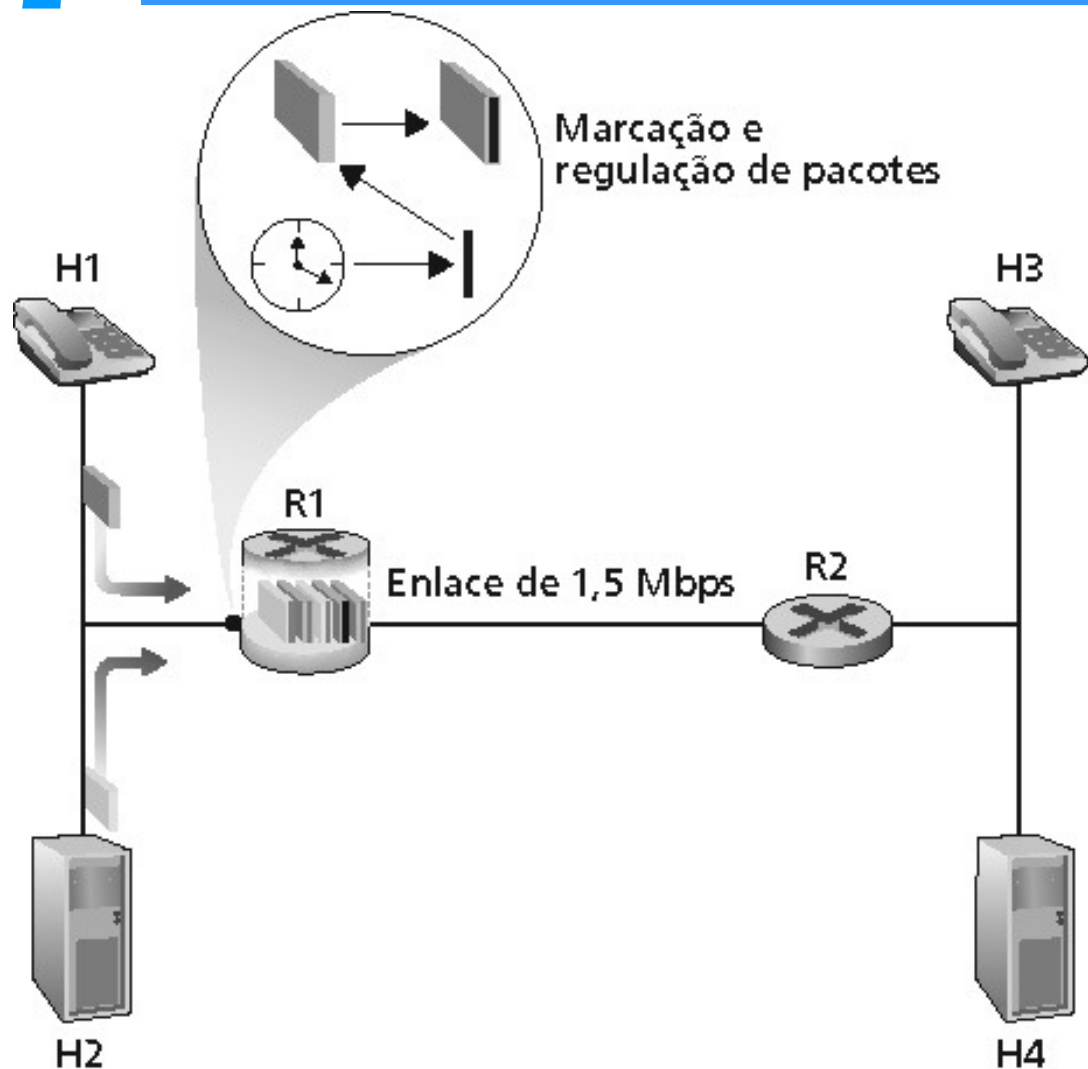


PEARSON

Addison  
Wesley

## 7

# Princípios para garantias de QoS (mais)



- Aplicações malcomportadas (áudio envia pacotes numa taxa superior do que a taxa declarada)
  - Policiamento: força a fontes a aderirem às alocações de largura de banda
- Marcação e policiamento na borda da rede:
  - Similar ao ATM UNI (User Network Interface)

## Princípio 2

Fornecer proteção (*isolação*) para uma classe em relação às demais.

Legenda:



Medição e regulação



Marcas

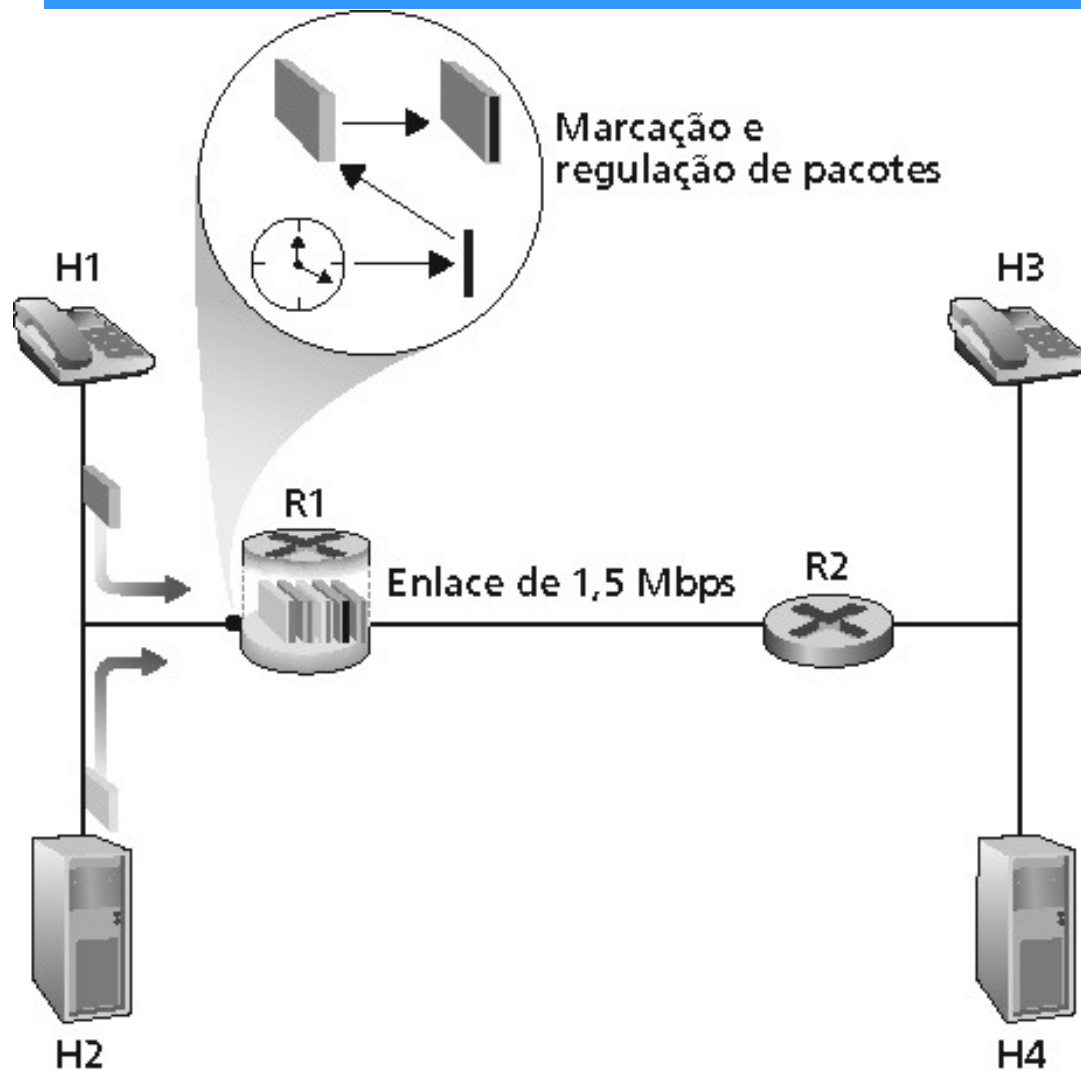


PEARSON

Addison  
Wesley

## 7

# Princípios para garantia de QoS (mais)



- Alocando largura de banda *fixa* (não compartilhável) para o fluxo: uso *ineficiente* da banda se o fluxo não usar sua alocação

## Princípio 3

Embora fornecendo isolamento, é necessário usar os recursos da forma mais eficiente possível.

Legenda:



Medição e regulação



Marcas



# 7

## Princípios para garantia de QoS (mais)

- *Fato básico de vida*: não pode suportar demandas de tráfego além da capacidade do enlace

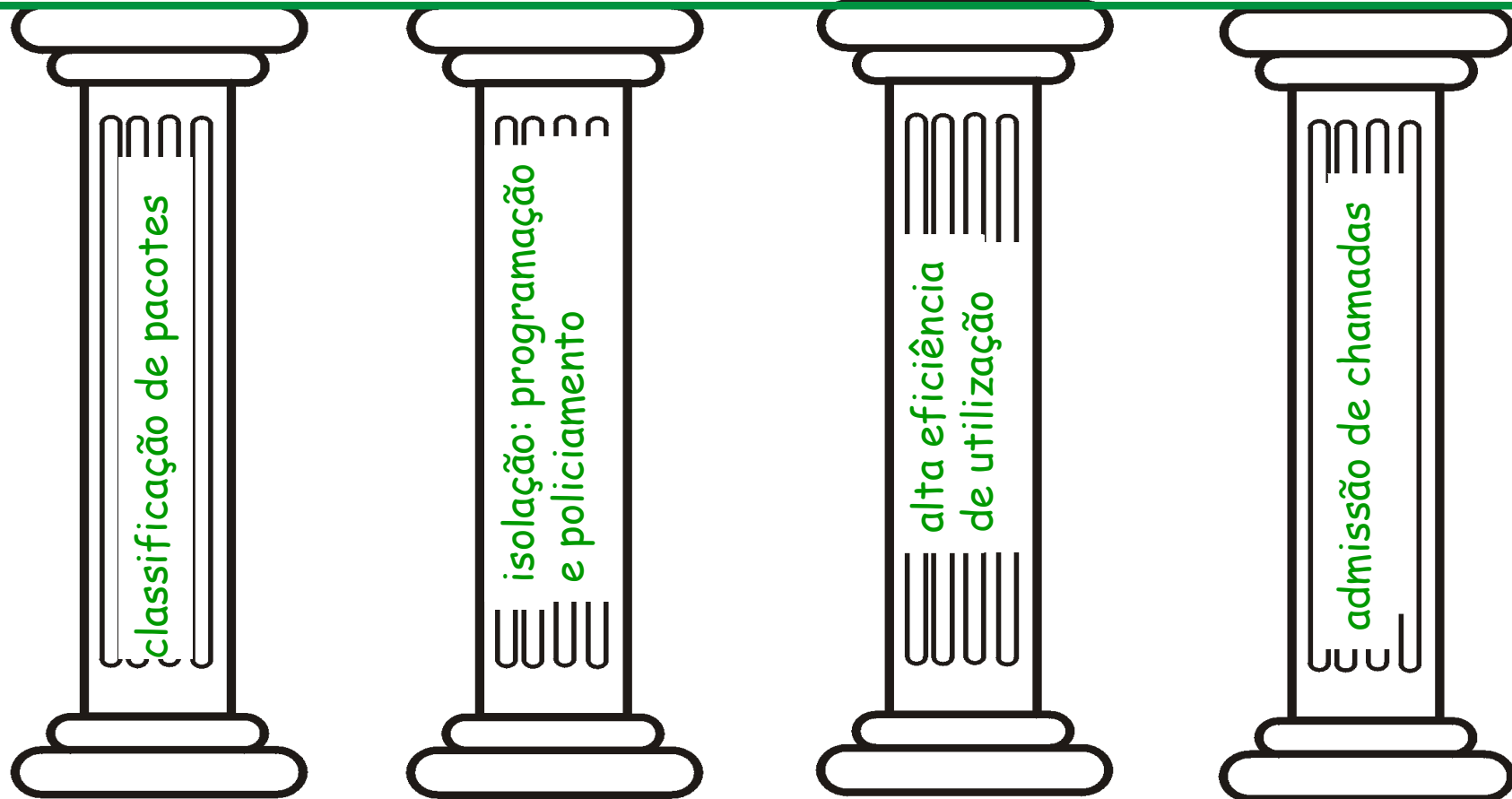
### Princípio 4

Admissão de chamada: fluxo declara suas necessidades, a rede pode bloquear a chamada (ex., sinal de ocupado) se ela não puder encontrar as necessidades.



# 7 Resumo dos princípios de QoS

## QoS para aplicações em redes



A seguir veremos os mecanismos para alcançar isso ....

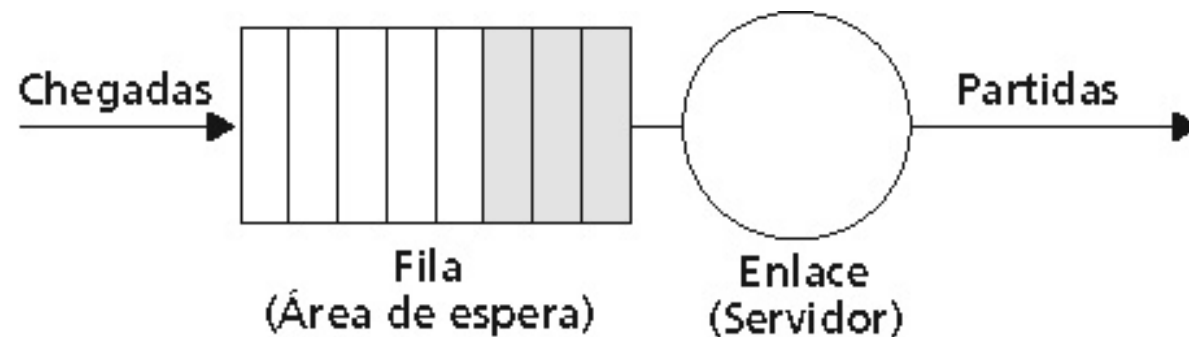
# 7 Redes multimídia

- 7.1 Aplicações de Rede Multimídia
- 7.2 Transmissão em fluxo contínuo de áudio e video armazenados
- 7.3 Multimídia em tempo real: estudo da Telefonia Internet
- 7.4 Protocolos para Aplicações Interativas em tempo real RTP, RTCP, SIP
- 7.5 Distribuição de Multimídia: redes de distribuição de conteúdo
- 7.6 Além do Melhor Esforço
- 7.7 Agendamento e Mecanismos de Policiamento
- 7.8 Serviços Integrados e Serviços Diferenciados
- 7.9 RSVP

# 7

## Mecanismos de escalonamento e policiamento

- **Escalonamento:** escolhe o próximo pacote para enviar no enlace
- **Escalonamento FIFO (first in first out):** envia em ordem de chegada para fila
- Exemplo do mundo real?
  - **Política de descarte:** se o pacote chega à fila cheia: quem descartar?
    - Descarta extremidade: descarta pacote que chega
    - Prioridade: descarta/remove com base em prioridade
    - Aleatório: descarta/remove aleatoriamente

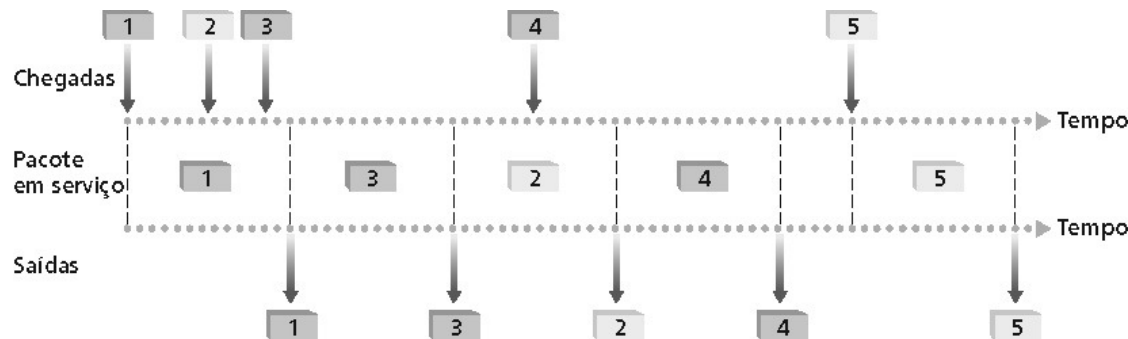
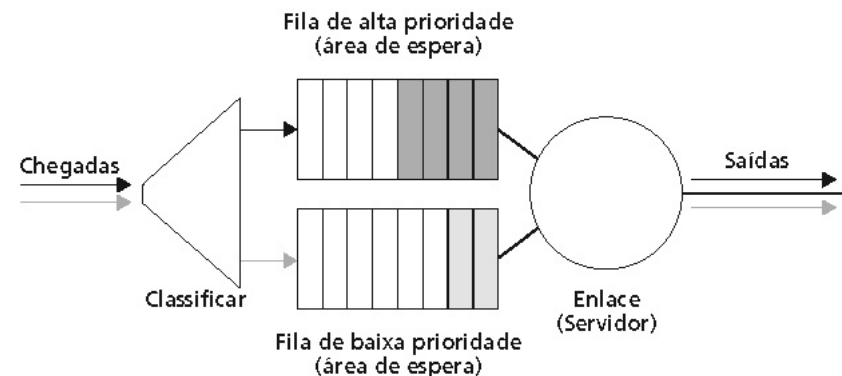


# 7

## Políticas de escalonamento e policiamento (cont.)

**Filas com prioridade:** transmite um pacote de prioridade mais alta que esteja presente na fila

- Múltiplas *classes*, com diferentes prioridades
  - Classes podem depender de marcação explícita ou de outras informações no cabeçalho. Ex., o IP de origem/destino, números de porta etc.
  - Exemplo do mundo real?

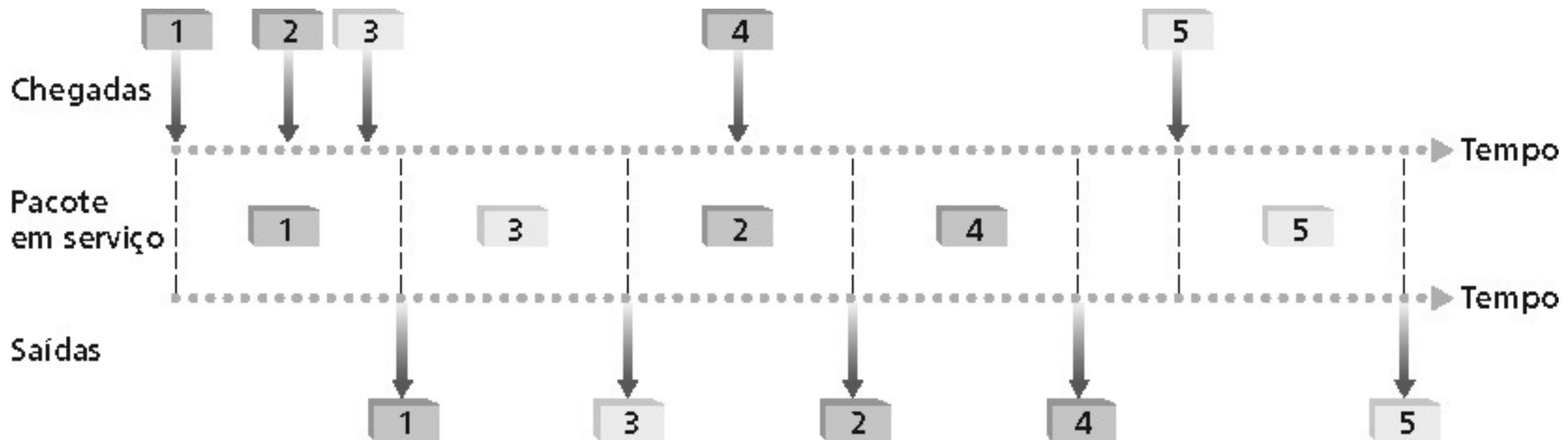


# 7

## Políticas de escalonamento e policiamento (cont.)

### Escalonamento round robin:

- Múltiplas classes
- Ciclicamente percorre as classes presentes na fila, servindo um pacote de cada classe (se disponível)
- Exemplo do mundo real?

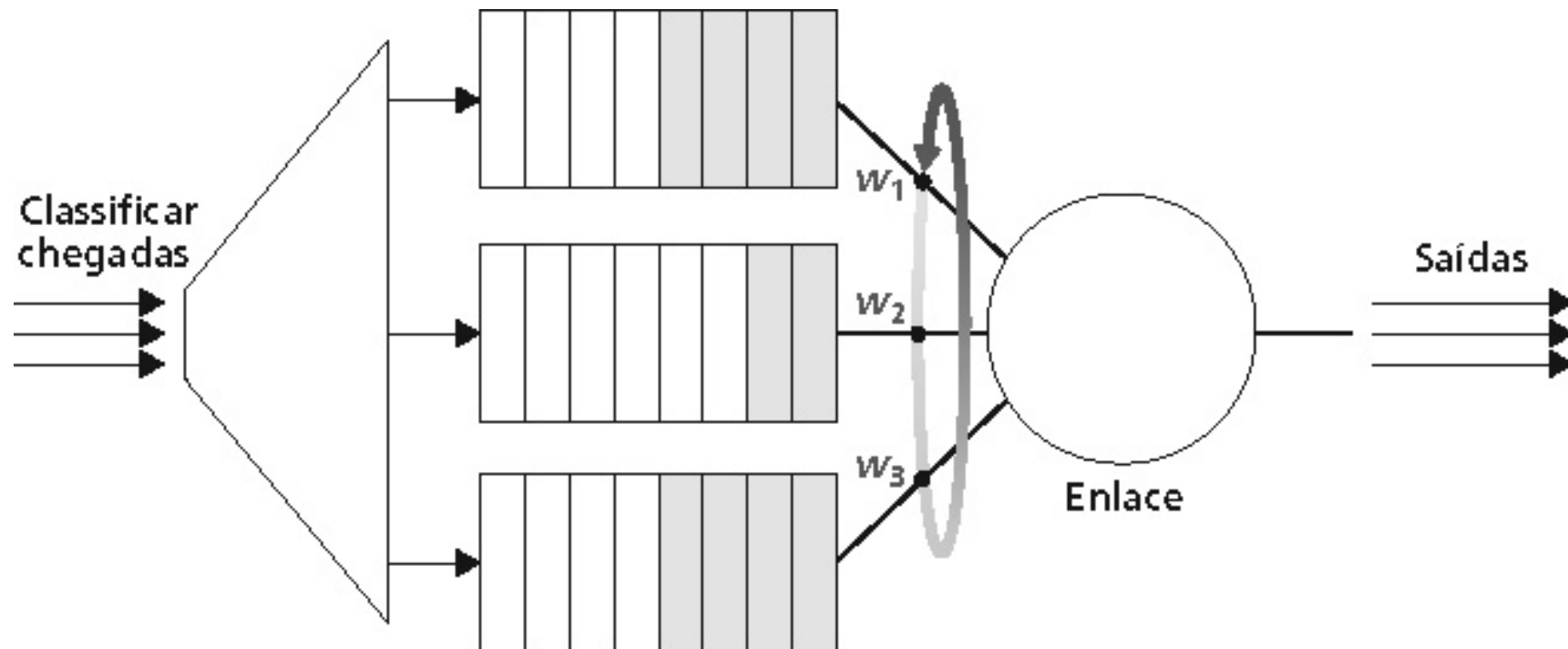


PEARSON

Addison  
Wesley

## Weighted Fair Queuing (Fila justa ponderada):

- Forma generalizada de Round Robin
- Cada classe obtém um volume diferenciado de serviço em cada ciclo
- Exemplo do mundo real?



# 7 Mecanismos de policiamento

**Objetivo:** limitar o tráfego para não exceder os parâmetros declarados

Três critérios comuns utilizados:

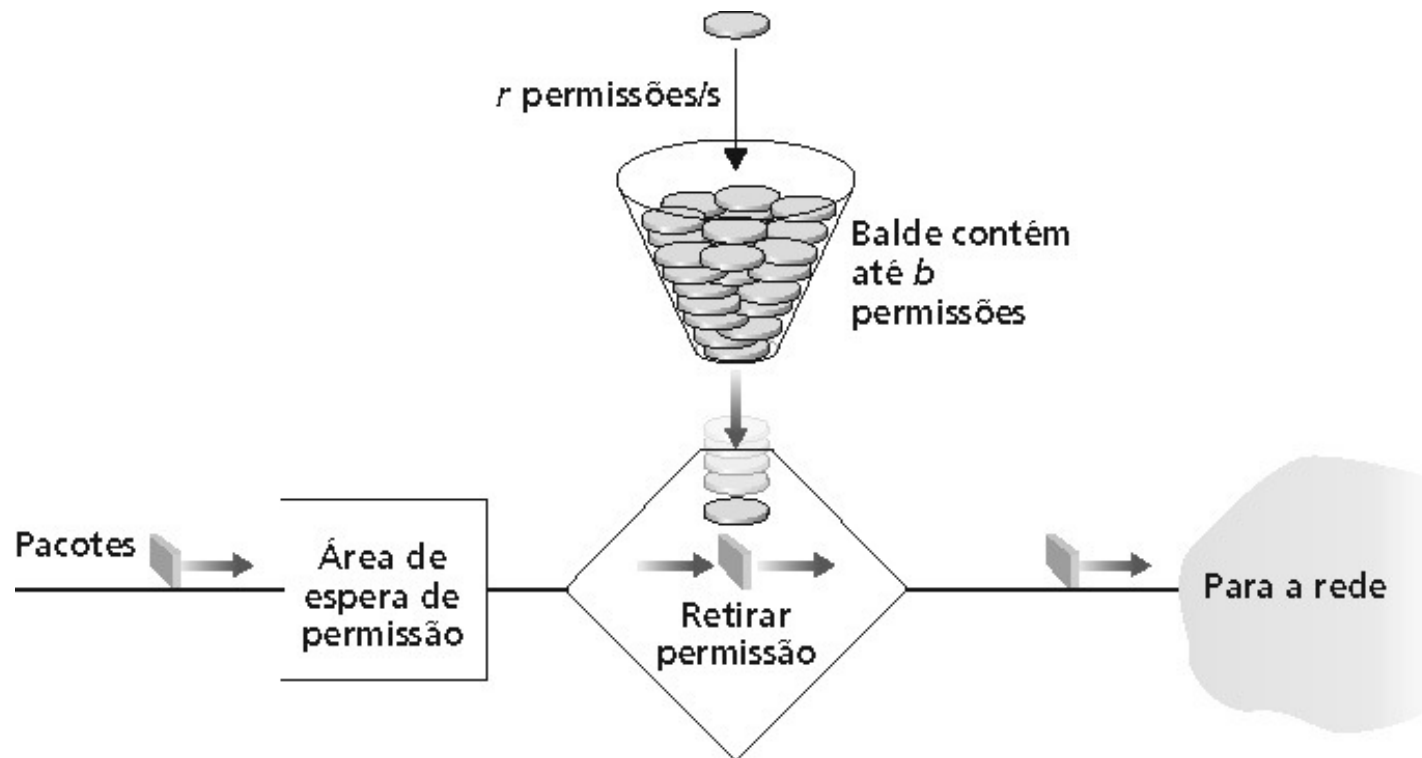
- *(longo prazo) taxa média:* quantos pacotes podem ser enviados por unidade de tempo? (a longo prazo)
  - questão crucial: qual é o tamanho do intervalo: 100 pacotes por segundo ou 6.000 pacotes por minuto possuem a mesma média!
- *taxa de pico:* ex., 6.000 pacotes por minuto (ppm) na média; taxa de pico de 1.500 ppm
- *(max.) tamanho da rajada:* número máximo de pacotes enviados consecutivamente (num curto período de tempo)





# 7 Mecanismos de policiamento

**Token bucket:** limita a entrada dentro de um tamanho de rajada e uma taxa média especificados.



- Bucket pode segurar  $b$  tokens
- Tokens gerados numa taxa  $r$  token/s exceto se o balde está cheio
- *Num intervalo de tamanho  $t$ : número de pacotes admitidos menor ou igual a  $(r t + b)$ .*

# 7 Mecanismos de policiamento

- Token bucket e WFQ podem ser combinados para prover um limite superior ao atraso, ex., *garantia de QoS!*



# 7 Redes multimídia

- 7.1 Aplicações de Rede Multimídia
- 7.2 Transmissão em fluxo contínuo de áudio e video armazenados
- 7.3 Multimídia em tempo real: estudo da Telefonia Internet
- 7.4 Protocolos para Aplicações Interativas em tempo real RTP, RTCP, SIP
- 7.5 Distribuição de Multimídia: redes de distribuição de conteúdo
- 7.6 Além do Melhor Esforço
- 7.7 Agendamento e Mecanismos de Policiamento
- 7.8 Serviços Integrados e Serviços Diferenciados
- 7.9 RSVP

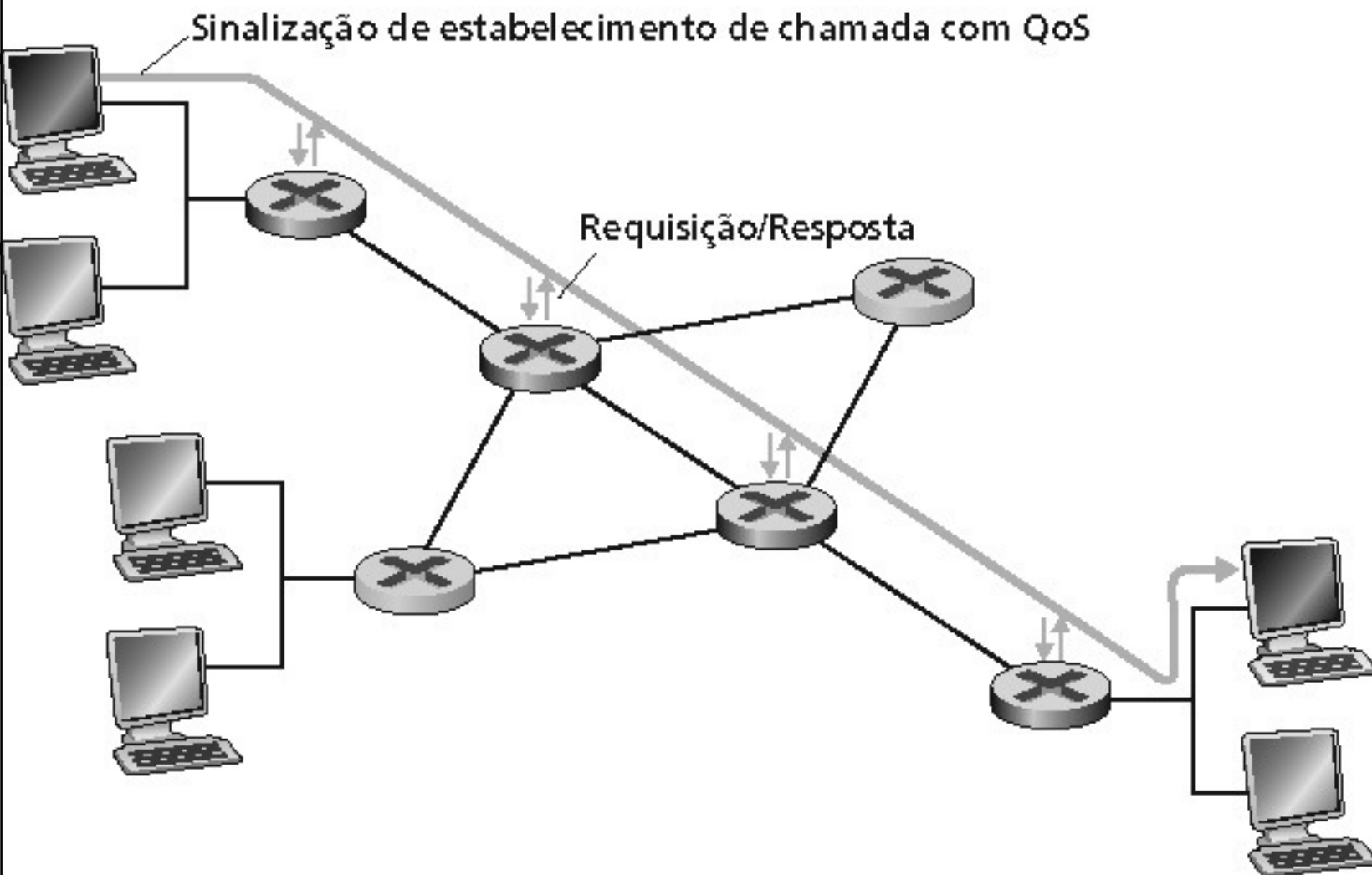
# 7 Serviços integrados do IETF

- Arquitetura para prover garantias de QoS em redes IP para sessões individuais de aplicações
- Reserva de recursos: roteadores mantêm informação de estado (como VC) dos recursos alocados, requisições de QoS
- Admite/recusa novas requisições de estabelecimento de chamada

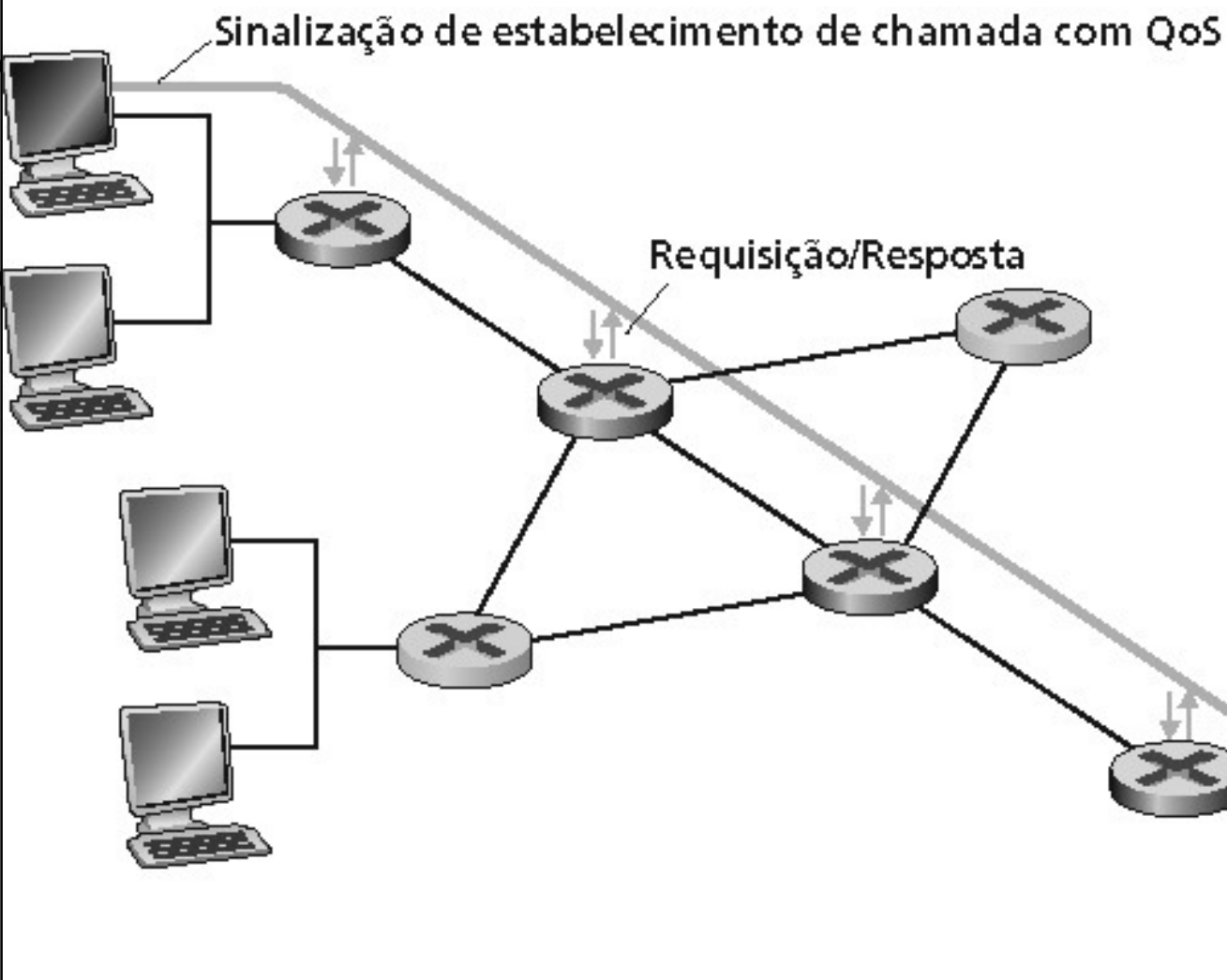
**Questão:** um fluxo recém-chegado pode ser admitido com garantias de desempenho enquanto não violar as garantias de QoS feitas aos fluxos já admitidos?



# 7 Intserv: cenário de garantia de QoS



# 7 Intserv: cenário de garantia de QoS



- Reserva de recursos
  - Estabelecimento de chamada, sinalização (RSVP)
  - Tráfego, declaração de QoS
  - Controle de admissão por elemento
  - Escalonamento sensível a QoS(ex., WFQ)

Na chegada, a sessão deve:

- Declarar seus requisitos de QOS
  - **R-spec**: define a QOS que está sendo solicitada
- Caracteriza o tráfego que será enviado pela rede
  - **T-spec**: define as características de tráfego
- Protocolo de sinalização: necessário para transportar um R-spec e um T-spec aos roteadores (quando reserva é necessária)
  - **RSVP**



## Serviço garantido:

- Pior caso de chegada de tráfego: fonte policiada por furado
- Simples (matematicamente demonstrável) *limitado* no atraso [Parekh 1992, Cruz 1988]

## Serviço de carga controlada:

- "a qualidade do serviço aproxima-se do QoS que o mesmo fluxo receberia de um elemento da rede sem carga."





# 7 Serviços diferenciados da IETF

## Preocupações com Intserv:

- **Escalabilidade:** sinalização, mantendo difícil o estado do roteador por fluxo com grande número de fluxos
- **Modelos de serviço flexíveis:** Intserv possui apenas duas classes. Também quer classes de serviço “qualitativas”
  - “comporta-se como um fio”
  - Distinção de serviço relativo: Platina, Ouro, Prata

## Abordagem Diffserv:

- Funções simples no núcleo da rede, funções relativamente complexas nos roteadores de borda (ou hospedeiros)
- Não define classes de serviço, fornece componentes funcionais para construir classes de serviço

# 7 Arquitetura Diffserv

## Roteador de borda:

- Gerenciamento de tráfego por fluxo
- Marca os pacotes com no perfil e fora de perfil

## Roteador de núcleo:

- Gerenciamento de tráfego por classe
- Armazenamento em buffer e escalonamento baseado na marcação na borda
- Preferência dada aos pacotes no perfil
- Encaminhamento garantido

- **Perfil:** taxa A pré-negociada, tamanho do balde B
- Marcação do pacote na borda baseada no perfil **por fluxo**

## Uso possível de marcação:

- Marcação baseada em classe: pacotes de diferentes classes marcados de modo diferente
- Marcação intraclasses: porção de fluxo em conformidade marcada de modo diferente que daqueles em não conformidade



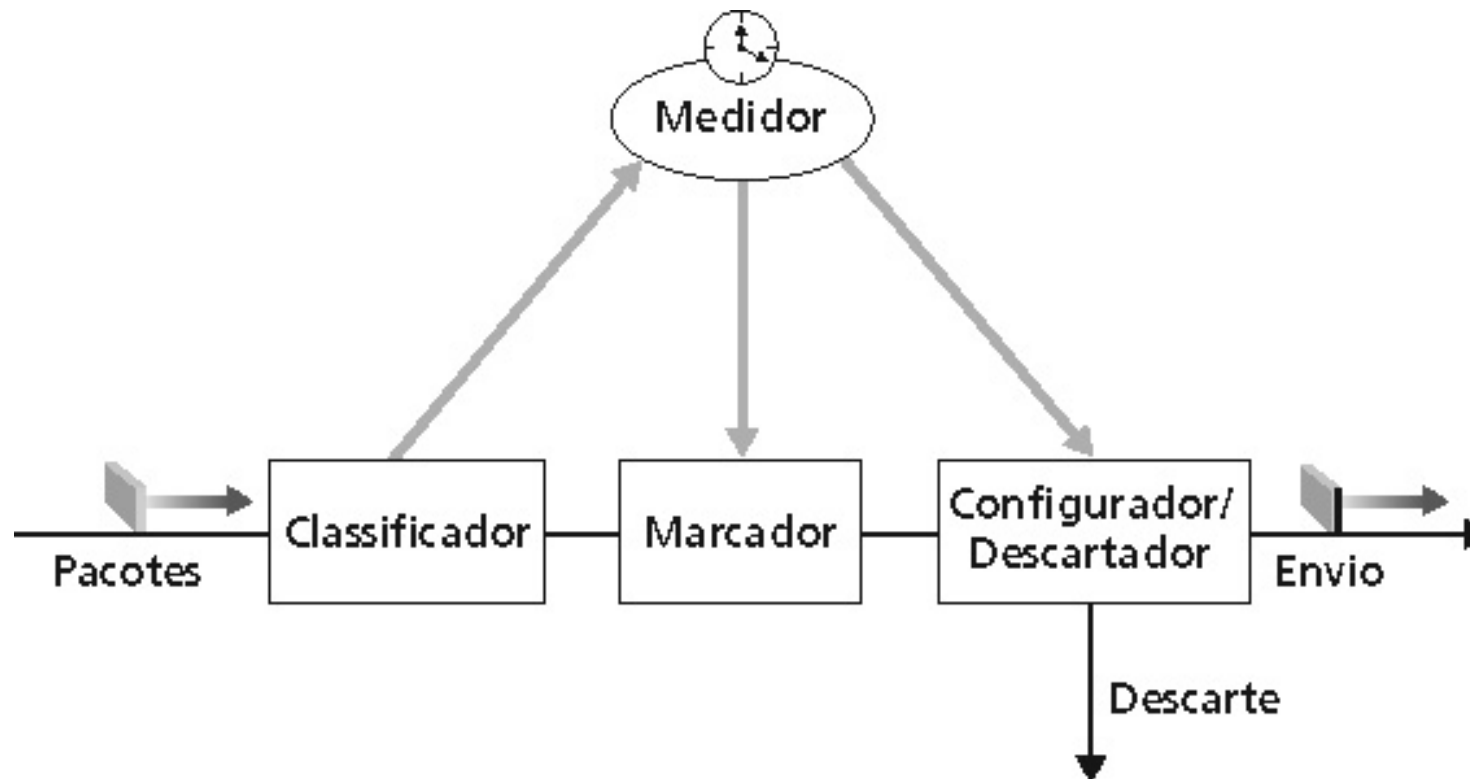
# 7 Classificação e condicionamento

- Pacote é marcado no campo Tipo de Serviço (TOS) no IPv4, e Classe de Tráfego no IPv6
- 6 bits são usados para o Ponto de Código de Serviços Diferenciados (DSCP) - (Differentiated Service Code Point) e determinam o PHB que o pacote receberá
- 2 bits são atualmente não usados



# 7 Classificação e condicionamento

- Pode ser desejável limitar a taxa de injeção de tráfego em alguma classe;
- Usuário declara o perfil de tráfego (ex., taxa e tamanho das rajadas)
- Tráfego é medido e ajustado se não estiver de acordo com o seu perfil



- PHB resulta num comportamento observacio-nalmente diferente (mensurável) para o desempenho do envio de pacotes
- PHB não especifica quais mecanismos usar para assegurar um comportamento do desempenho conforme o exigido pelo PHB
- Exemplos:
  - Classe A obtém  $x\%$  da taxa de transmissão do enlace de saída considerando intervalos de tempo de uma certa extensão
  - Pacotes de classe A partem primeiro, antes dos pacotes de classe B

- PHBs que estão sendo estudados:
  - **Envio expresso:** taxa de partida dos pacotes de uma dada classe iguala ou excede uma taxa especificada
    - Enlace lógico com uma taxa mínima garantida
  - **Envio assegurado:** 4 classes de tráfego
    - Cada uma garantida com um mínimo de largura de banda;
    - Cada uma com três particionamentos para preferência de descarte dos pacotes

# 7 Redes multimídia

- 7.1 Aplicações de Rede Multimídia
- 7.2 Transmissão em fluxo contínuo de áudio e video armazenados
- 7.3 Multimídia em tempo real: estudo da Telefonia Internet
- 7.4 Protocolos para Aplicações Interativas em tempo real RTP, RTCP, SIP
- 7.5 Distribuição de Multimídia: redes de distribuição de conteúdo
- 7.6 Além do Melhor Esforço
- 7.7 Agendamento e Mecanismos de Policiamento
- 7.8 Serviços Integrados e Serviços Diferenciados
- 7.9 RSVP



# 7 Sinalização na Internet

Encaminhamento sem  
conexão (sem estado)  
pelos roteadores IP + serviço de melhor  
esforço = sem protocolos de  
sinalização no  
projeto inicial do IP

- **Novo requisito:** reserva de recursos ao longo do caminho fim-a-fim (sistema final, roteadores) para QoS de aplicações multimídia
- **RSVP:** Resource Reservation Protocol [RFC 2205]
  - “ ... permite que os usuários comuniquem suas exigências à rede de maneira robusta e eficiente.” Ex., sinalizando!
- Protocolo mais antigo de sinalização na Internet: ST-II [RFC 1819]

# 7 Objetivos do projeto RSVP

1. Acomodar **receptores heterogêneos** (largura de banda diferente ao longo do caminho)
2. Acomodar diferentes aplicações **com diferentes requisitos de recursos**
3. Fazer **multicast de um serviço de primeira classe**, com adaptação para sociedade de grupo de multicast
4. **Alavancar o roteamento multicast/unicast existente**, com adaptação às mudanças nas rotas de unicast/multicast
5. **Cabeçalho do protocolo de controle com** crescimento (no pior caso) linear do número de receptores
6. **Projeto modular** para tecnologias heterogêneas

# 7 RSVP: não...

- Especifica como os recursos estão para serem reservados
  - Prefere: um mecanismo para necessidades de comunicação
- Determina rotas que os pacotes seguirão
  - Isso é trabalho para protocolos de roteamento
  - Sinalização desacoplada do roteamento
- Interage com encaminhamento de pacotes
  - Separação de controle (sinalização) e planos de dados (encaminhamento)

# 7 RSVP: visão geral da operação

- Transmissor/receptor juntam-se a um grupo de multicast
  - Feito fora do RSVP
  - Transmissor não precisa se juntar a um grupo
- Sinalização do transmissor para rede
  - *Mensagem de caminho*: faz a presença do transmissor ser conhecida pelos roteadores
  - Abandonar caminho: apaga dos roteadores o estado do caminho do transmissor
- Sinalização do receptor para rede
  - *Mensagem de reserva*: reserva recursos do transmissor para o receptor
  - Abandonar reserva: remove reservas do receptor
- Sinalização da rede para sistema final
  - Erro de caminho
  - Erro de reserva



# 7 Mensagens de caminho: sinalização RSVP *transmissor para rede*

- Conteúdo da **mensagem de caminho**:
  - **Address**: destino do unicast, ou grupo de multicast
  - **Flowspec**: especifica requisitos de banda
  - **Filter flag**: se sim, grava identidades dos emissores de upstream (para permitir filtragem de pacotes por origem)
  - **Previous hop**: ID do roteador/host de upstream
  - **Refresh time**: tempo até esta informação expirar
- Mensagem de caminho: comunica informações do transmissor e informações de roteamento de caminho-reverso-para-transmissor
  - Encaminhamento posterior para upstream das reservas do receptor

# 7

## RSVP: conferência de áudio simples

- H1, H2, H3, H4, H5 são todos transmissores e receptores
- Grupo de multicast  $m1$
- Sem filtragem: pacotes de qualquer transmissor encaminhadas
- Taxa de áudio:  $b$
- Apenas uma árvore de roteamento multicast possível



PEARSON

Addison  
Wesley

# 7

## RSVP: construindo estado de caminho

- H1, ..., H5 todos enviam mensagens de caminho em *m1*:  
(address = *m1*, Tspec = *b*, filter-spec = no-filter, refresh = 100)
- Supõe que que H1 envia a primeira mensagem de caminho



# 7

## RSVP: construindo estado de caminho (cont.)

- Em seguida, H5 envia mensagem de caminho, criando mais estados nos roteadores



PEARSON

Addison  
Wesley



# 7

## RSVP: construindo estado de caminho (cont.)

- H2, H3, H5 enviam mensagens de caminho, completando as tabelas de estado de caminho



- Mensagem de reserva contém:
  - *Largura de banda desejada:*
  - *Tipo de filtro:*
    - **Sem filtro:** qualquer endereço de pacote para um grupo multicast pode utilizar reserva
    - **Filtro fixo:** apenas pacotes de um grupo específico de transmissores podem utilizar reserva
    - **Filtro dinâmico:** transmissores cujos pacotes podem ser encaminhados pelo enlace mudarão (por escolha do receptor) através do tempo.
  - *Filter spec*
- Fluxos upstream de reservas do receptor para transmissores, reservando recursos, criando estados adicionais *relacionados ao receptor* em roteadores

# 7

## RSVP: reserva do *receptor* (exemplo 1)

H1 quer receber áudio de todos os outros transmissores

- Mensagem de reserva do H1 flui árvore acima para as origens
- H1 reserva apenas largura de banda suficiente para um fluxo de um áudio
- Reserva é do tipo “sem filtro” - qualquer transmissor pode usar largura de banda reservada



# 7

## RSVP: reserva do *receptor* (exemplo 1)

- Mensagens de reserva do H1 fluem árvore acima para as origens
- Roteadores, hospedeiros reservam largura de banda  $b$  necessária aos enlaces de downstream em direção ao H1



# 7 RSVP: reserva do *receptor* (mais)

- Em seguida, H2 faz reserva sem filtragem para banda  $b$
- H2 encaminha para R1, R1 encaminha para H1 e R2 (?)
- R2 não faz nada, pois  $b$  já está reservado em L6



# 7

## RSVP: reserva do *receptor*: características

E se houver múltiplos transmissores (e.g., H3, H4, H5) sobre o enlace (ex., L6)?

- Interpolação arbitrária de pacotes
- Fluxo L6 policiado por balde furado: se a taxa de transmissão de  $H3 + H4 + H5$  ultrapassar  $b$ , ocorre perda de pacote



# 7 RSVP: exemplo 2

- H1, H4 são apenas transmissores
  - Envia **mensagens de caminho** como antes, indicando reserva filtrada
  - Roteadores armazenam transmissores de upstream para casa enlace de upstream
- H2 vai querer receber de H4 (apenas)

# 7 RSVP: exemplo 2

- H1, H4 são apenas transmissores
  - Enviam **mensagens de caminho** como antes, indicando reserva filtrada



# 7 RSVP: exemplo 2

- Receptor H2 envia mensagem de reserva para origem H4 na banda  $b$ 
  - Upstream propagado em direção a H4, reservando  $b$

# 7 RSVP: *soft-state*

- Transmissores periodicamente reenviam mensagens de caminho para manter (*refresh*) o estado
- Receptores periodicamente reenviam mensagens de caminho para manter (*refresh*) o estado
- Mensagens de caminho e de reserva possuem campo TTL, especificando intervalo de *refresh*



# 7 RSVP: *soft-state*

- Suponha que H4 (transmissor) permanece sem encerrar a conexão
- Finalmente o estado nos roteadores irá expirar e desaparecer!



# 7

## Vários usos de *refresh* de reserva/caminho

- Recuperar-se de uma perda anterior de mensagem *refresh*
- Tempo esperado até a recepção do *refresh* deve ser maior do que o intervalo de expiração! (menor intervalo de tempo desejado)
- Manipular receptor/transmissor que sai sem encerrar a conexão
  - Estado do transmissor/receptor irá expirar e desaparecer
- *Refreshes* de reserva farão com que novas reservas sejam feitas para um receptor de um transmissor que se juntou desde o último refresh de reserva dos receptores
- Ex., no exemplo anterior, H1 é apenas receptor, H3 apenas transmissor. Mensagens de caminho/reserva se completam, os dados fluem
  - H4 se junta como transmissor, nada acontece até que H3 realize o refresh da reserva, fazendo com que R3 encaminhe a reserva para H4, que irá alocar a banda



# 7 RSVP: reflexões

- Multicast como um serviço de “primeira classe”
- Reservas orientadas ao receptor
- Uso de soft-state

# 7 Redes multimídia: resumo

- Aplicações e requisitos de multimídia
- Obtendo o melhor do serviço de melhor esforço atual
- Mecanismos de escalonamento e policiamento
- Internet da próxima geração: Intserv, RSVP, Diffserv