

# Mestrado Profissional em Computação

Uece-IFCE

Disciplina: Protocolos

*Chico Anysio*  
*Óleo sobre tela 60x40 cm*

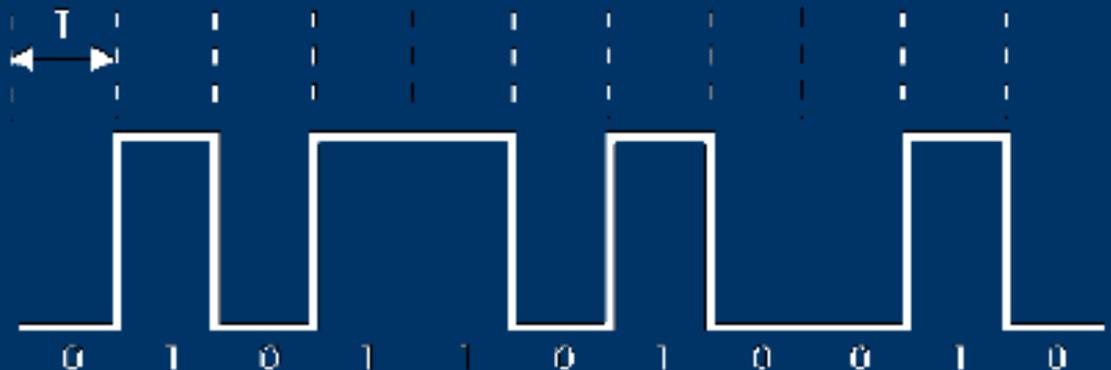


# Tipos de Sinal

- Analógico
  - Variação contínua



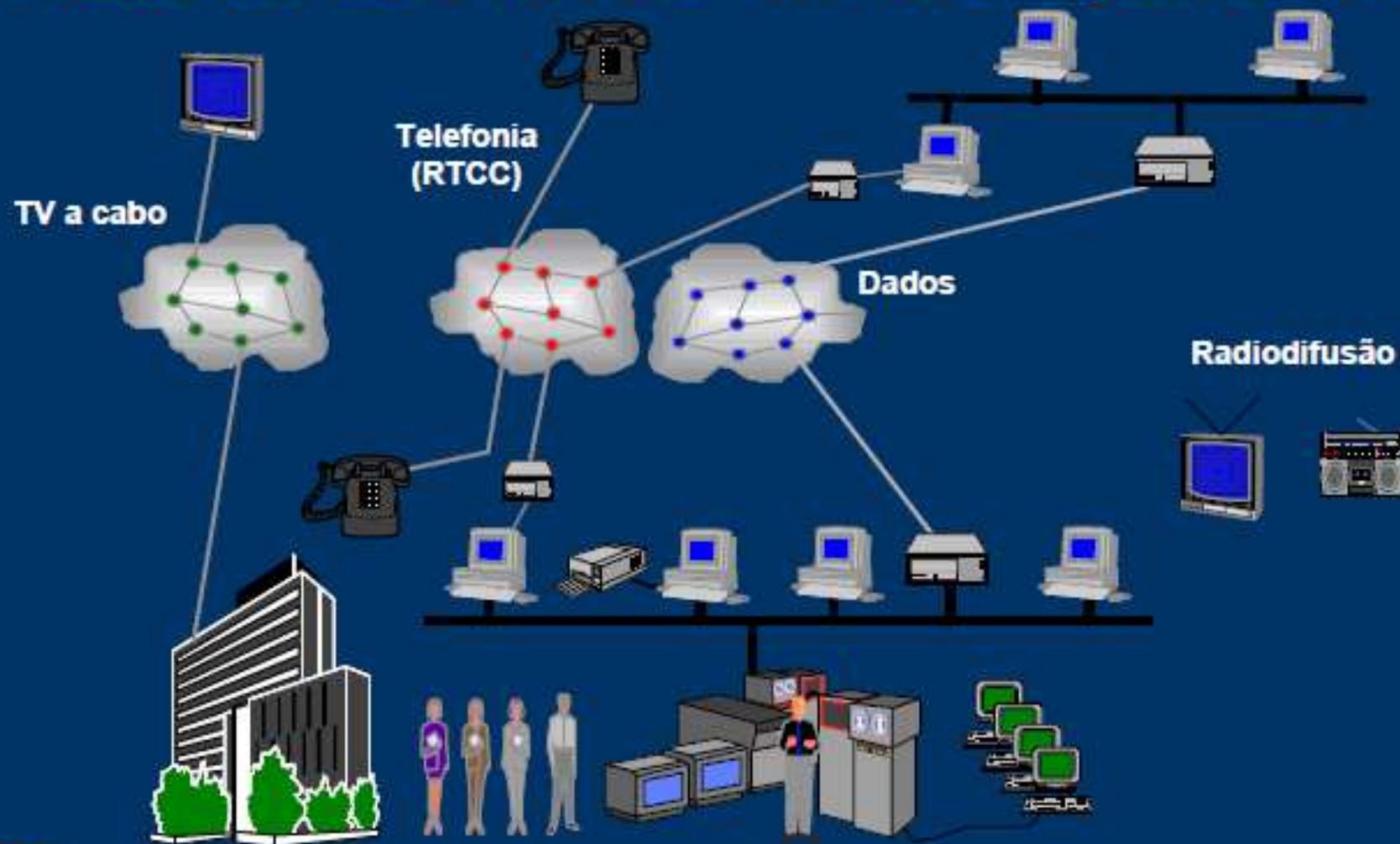
- Digital
  - Variação discreta
  - Intervalo (constante) de sinalização

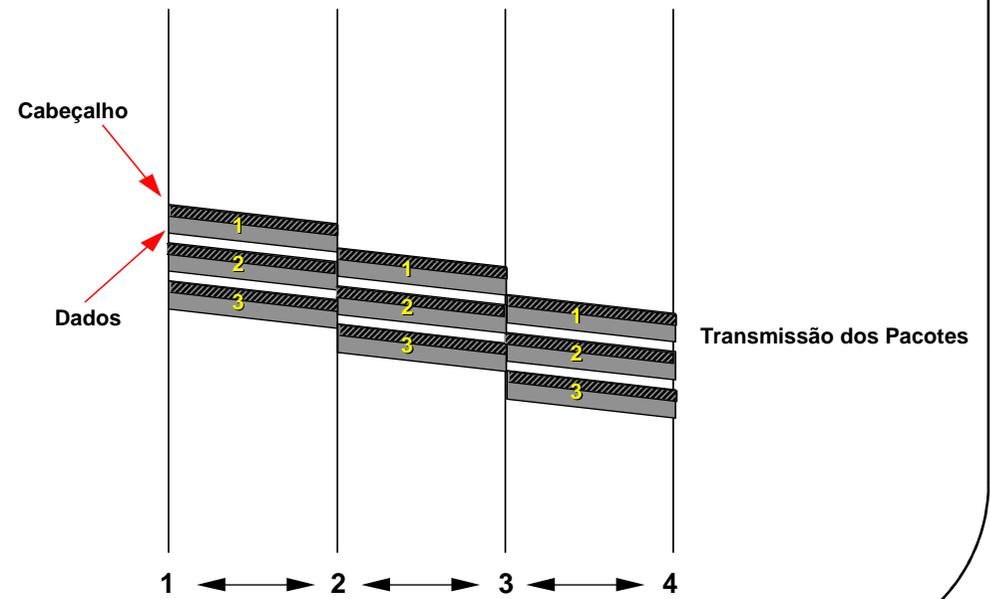
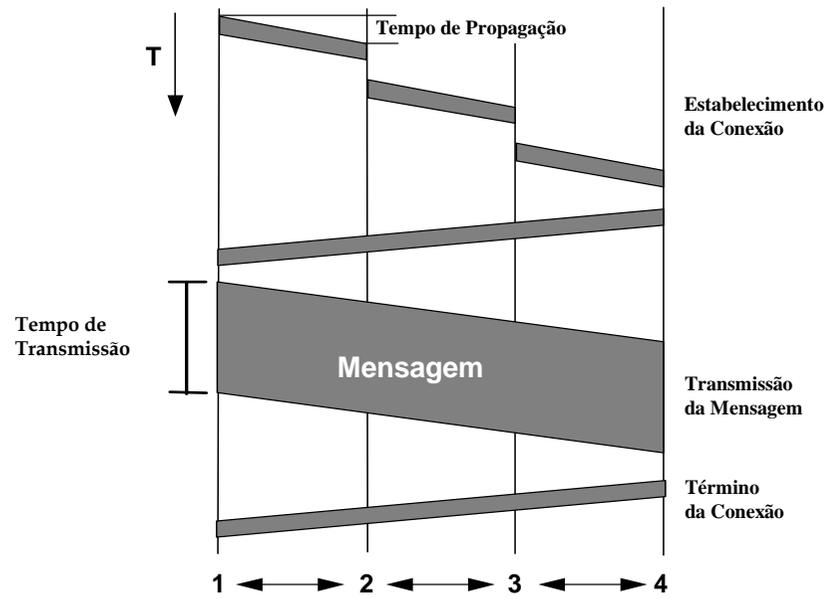


**Qualquer informação pode ser transmitida através de sinal analógico ou digital**

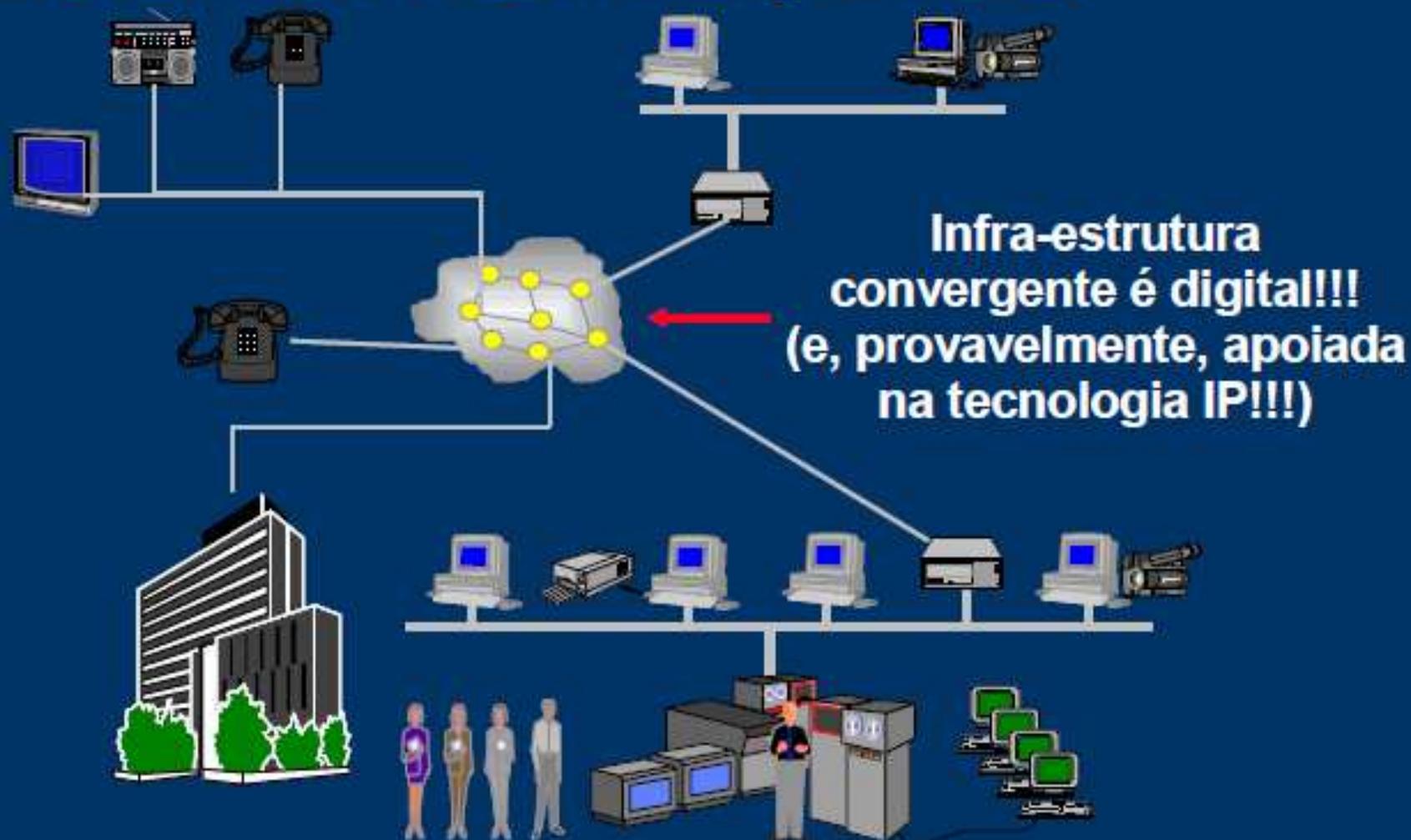


# Sistemas de Comunicação Multimídia (Não-convergentes)



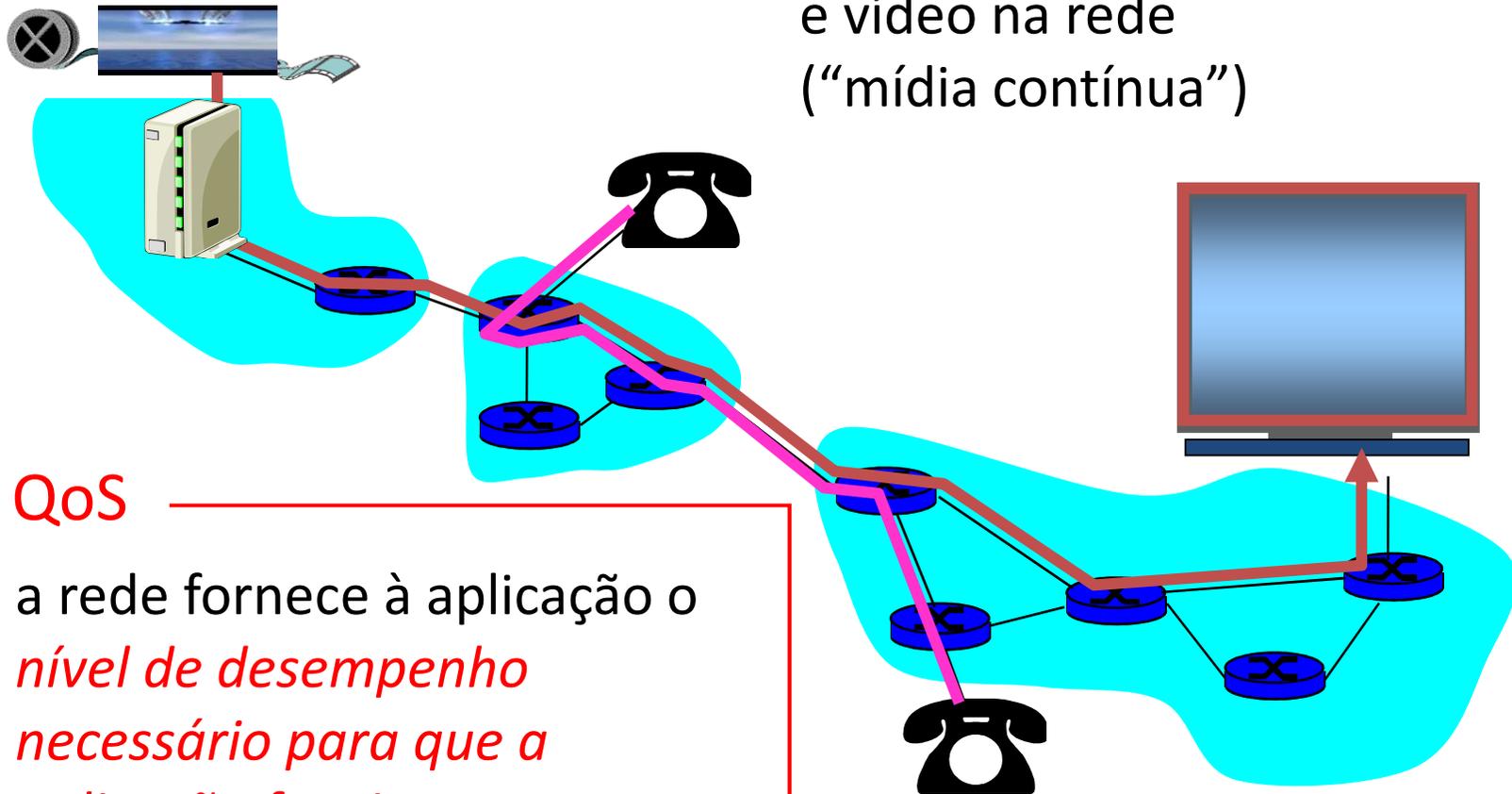


# Sistemas de Comunicação Multimídia (Convergentes)



# O que são Multimídia e Qualidade de Serviço?

**Aplicações Multimídia:** áudio e vídeo na rede (“mídia contínua”)

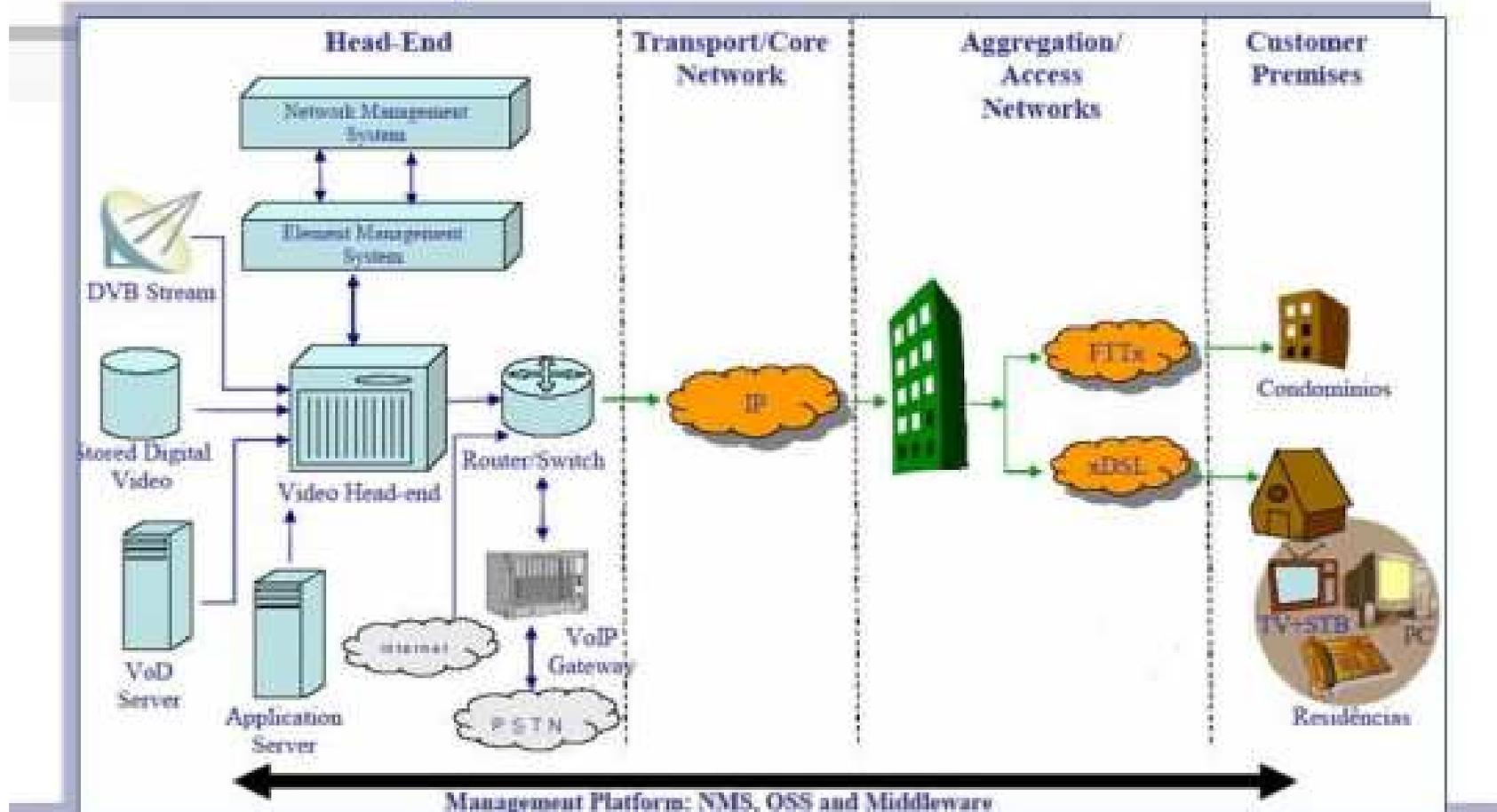


## QoS

a rede fornece à aplicação o *nível de desempenho necessário para que a aplicação funcione como esperado*

O que são Multimídia e Qualidade de Serviço?

## IPTV – Arquitectura de Rede



# Aplicações de Rede Multimídia

## Classes de aplicações de Multimídia:

- 1) Áudio e vídeo de fluxo contínuo **armazenados** (*Streams*)
- 2) Áudio e vídeo de fluxo contínuo **ao vivo**
- 3) Áudio e vídeo **Interativos em tempo real**

**Jitter** é a variação de atraso dos pacotes dentro de um mesmo fluxo de pacotes

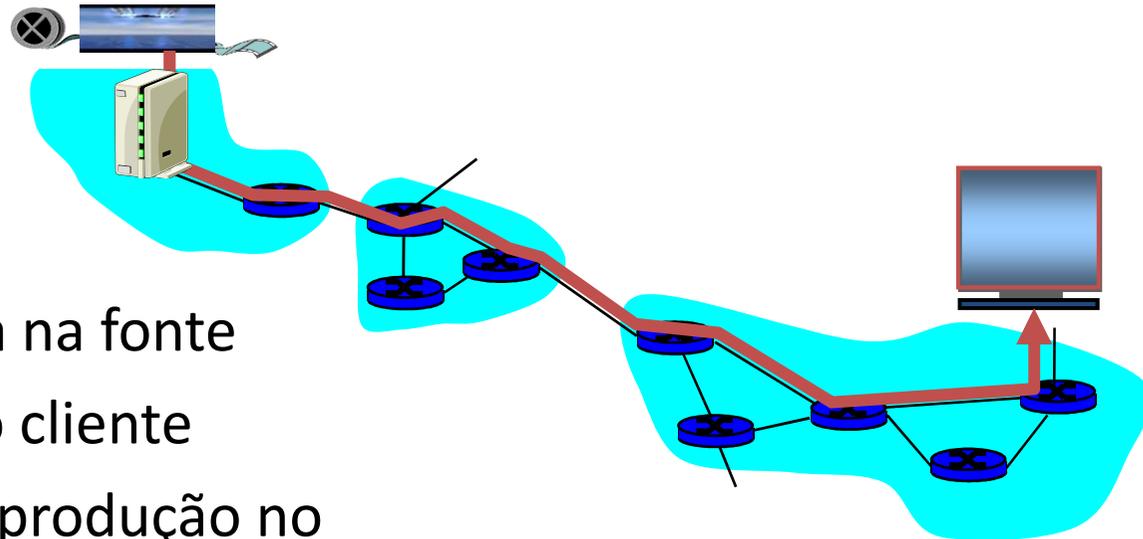
## Características Fundamentais:

- Tipicamente são **sensíveis a atrasos**
  - atraso fim-a-fim
  - variação do atraso (*jitter*)
- Mas são **tolerantes a perdas**: perdas ocasionais causam somente pequenas perturbações
- Antítese da transferência de dados que é intolerante a perdas mas tolerante a atrasos.

# 1) Áudio e vídeo de fluxo Contínuo Armazenados

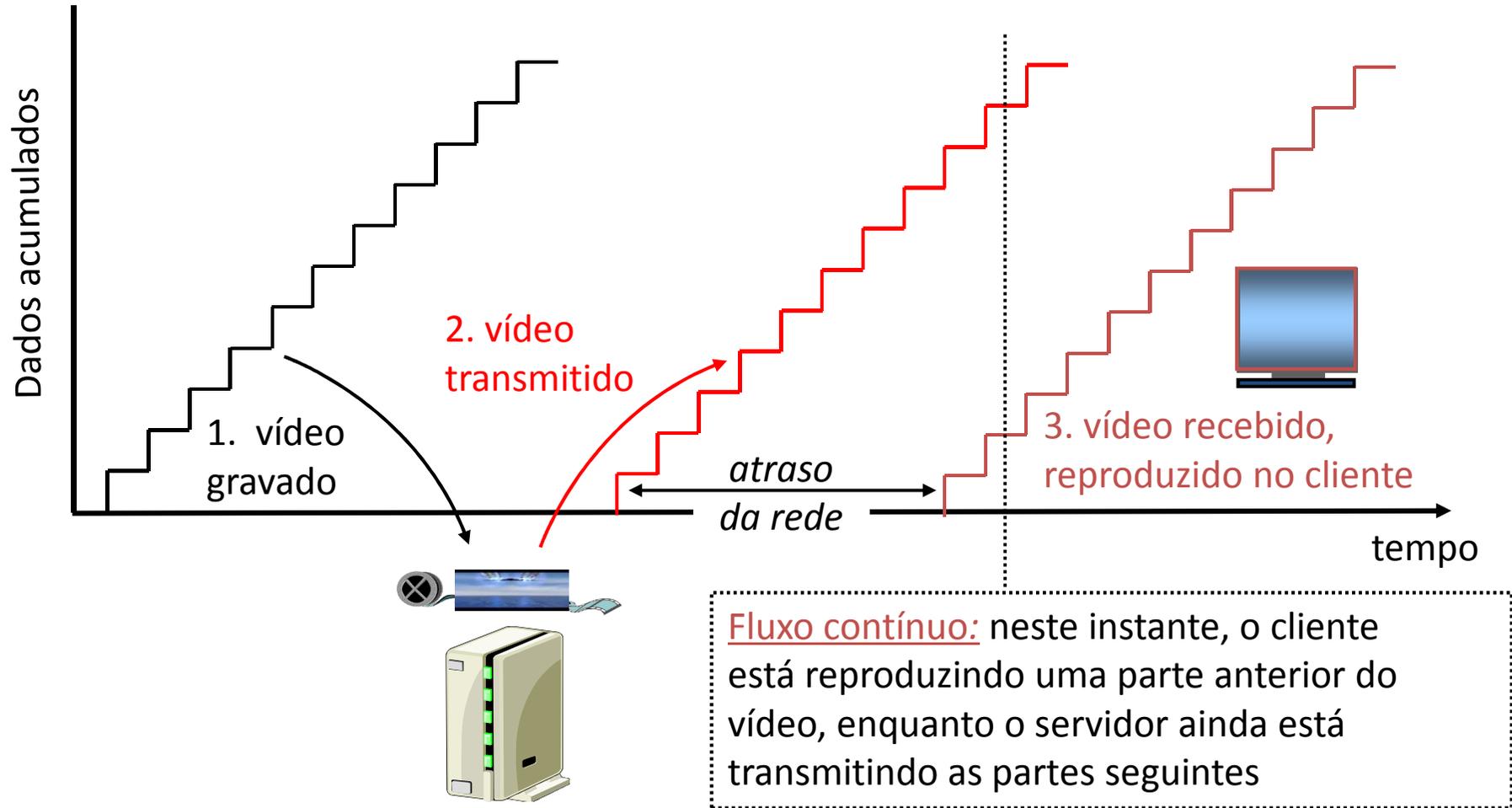
## Fluxo Contínuo (Streaming):

- ❑ mídia armazenada na fonte
- ❑ transmitida para o cliente
- ❑ Fluxo contínuo: reprodução no cliente inicia *antes* de que todos os dados tenham sido recebidos
- ❑ restrição de tempo para os dados ainda não transmitidos: devem chegar a tempo de serem reproduzidos



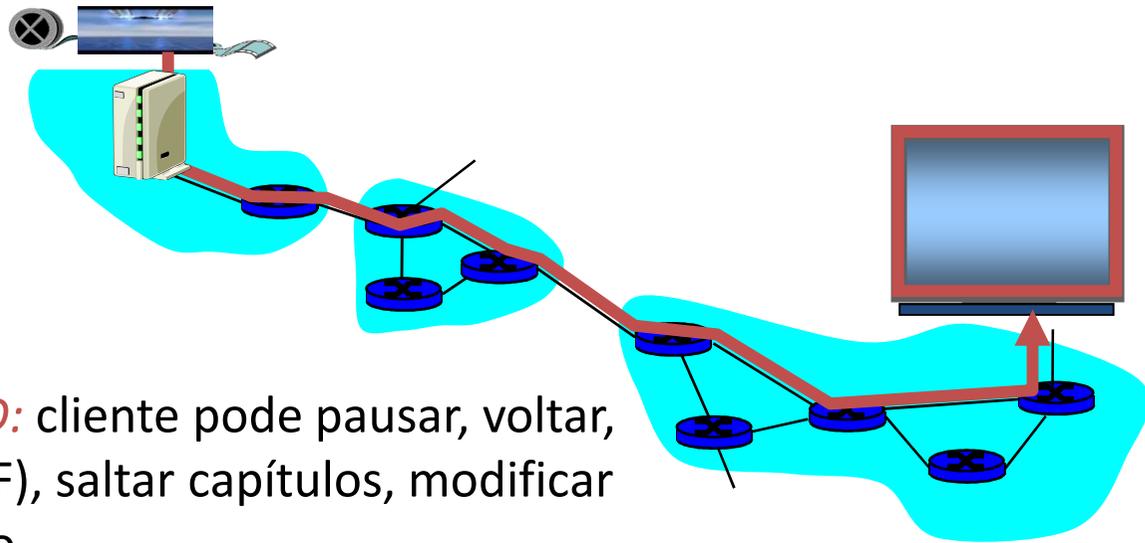
# 1) Áudio e vídeo de fluxo Contínuo Armazenados

Como funciona?



# 1) Áudio e vídeo de fluxo Contínuo Armazenados

Interatividade



❑ *Funcionalidade tipo DVD:* cliente pode pausar, voltar, avançar rapidamente (FF), saltar capítulos, modificar a barra de deslocamento

- atraso inicial de 10 seg OK
- 1-2 seg até que o comando seja executado OK
- RTSP é freqüentemente usado (mais detalhes posteriormente)

❑ restrição de tempo para dados ainda não transmitidos: chegar em tempo para reprodução

## 2) Áudio e vídeo de fluxo Contínuo ao vivo

### Exemplos:

- Programa de bate papo em rádio Internet
- Evento esportivo ao vivo

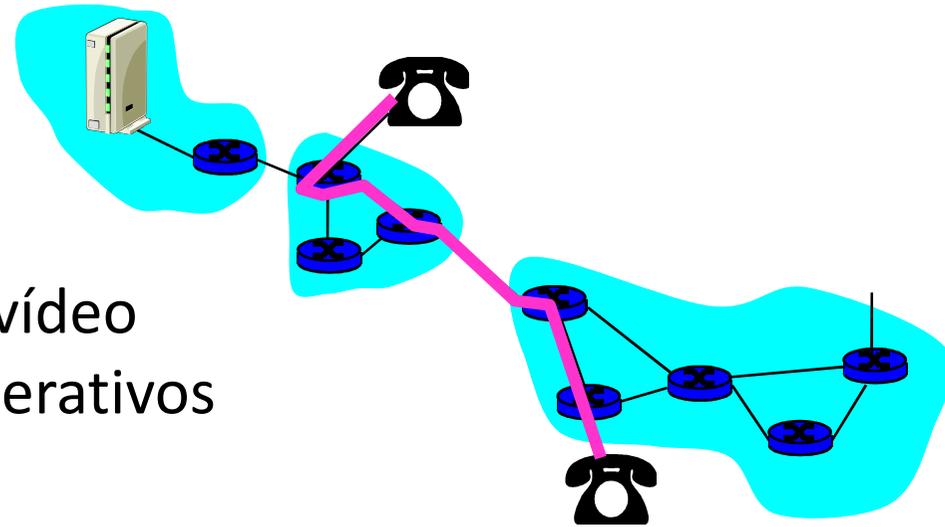
### Fluxo Contínuo

- buffer de reprodução
- reprodução pode atrasar dezenas de segundos após a transmissão
- ainda tem restrições de tempo

### Interatividade

- impossível avançar
- é possível voltar, pausar!

### 3) Áudio e vídeo interativos em tempo real



□ **aplicações:** telefonia IP, vídeo conferência, mundos interativos distribuídos

- **requisitos de atraso fim-a-fim:**

- áudio: < 150 mseg bom, < 400 mseg OK

- Inclui atrasos da camada de aplicação (empacotamento) e de rede
    - Grandes atrasos são perceptíveis, prejudicam a interatividade

- **inicialização da sessão**

- como o destino anuncia o seu endereço IP, número de porta e algoritmo de codificação?

# Multimídia Sobre a Internet Atual

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

- *sem* garantias sobre atrasos, perdas



?

?

?

?

?

?

?

Mas, você disse que as aplicações MM  
necessitam de QoS e nível de desempenho  
para funcionarem!

?

?

?



As aplicações MM na Internet atual usam técnicas da camada de aplicação para minimizar (da melhor forma) efeitos de atrasos e perdas

# Como a Internet deveria evoluir para dar melhor suporte à multimídia?

## Filosofia de serviços integrados:

- Modificar a Internet de modo que as aplicações possam reservar largura de banda fim-a-fim
- Requer software novo e complexo nos hospedeiros e roteadores

## Filosofia do “deixa como está”:

- sem grandes mudanças
- aumento da largura de banda quando necessário
- distribuição de conteúdo, multicast na camada de aplicação

## Filosofia de serviços diferenciados:

- Menos mudanças na infraestrutura da Internet, mas provendo serviços de 1a. e 2a. classes.



O que você acha?

# Algumas palavras sobre compressão de áudio

- Sinal analógico amostrado a uma taxa constante
  - telefone: 8.000 amostras/seg
  - CD de música: 44.100 amostras/seg
- Cada amostra é discretizada (arredondada): **quantização**
  - ex.,  $2^8=256$  possíveis valores discretos
- Cada valor de quantização é representado por bits
  - 8 bits para 256 valores
- Exemplo: 8.000 amostras/seg, 256 valores discretos --> 64.000 bps
- Receptor converte-o de volta a um sinal analógico:
  - alguma perda de qualidade

## Exemplo de taxas

- CD: 1,411 Mbps
- MP3 (MPEG 1 de camada 3): 96, 128, 160 kbps
- Telefonia Internet: 5,3 - 13 kbps

# Algumas palavras sobre compressão de vídeo

- Vídeo é uma seqüência de imagens apresentadas a uma taxa constante
  - ex. 24 imagens/seg
- Imagem digital é uma matriz de pixels
- Cada pixel é representado por bits que representam a luminância e cor
- Redundância
  - Espacial (dentro da imagem)
  - Temporal (de uma imagem para a próxima)

## Exemplos:

- MPEG 1 (CD-ROM) 1,5 Mbps
- MPEG2 (DVD) 3-6 Mbps
- MPEG4 (freqüentemente usado na Internet, < 1 Mbps)

## Pesquisa:

- vídeo em camadas (escalável)
  - adapta as camadas à largura de banda disponível

# Capítulo 7: Roteiro

- 7.1 Aplicações de Rede Multimídia
- 7.2 **Áudio e vídeo de fluxo contínuo armazenados**
- 7.3 Fazendo o melhor possível com o serviço de melhor esforço
  - 7.5 Distribuição de Multimídia: redes de distribuição de conteúdo
- 7.4 Protocolos para aplicações interativas em tempo real
  - RTP,RTCP,SIP
- 7.5 provendo múltiplas classes de serviço
  - 7.6 Além do melhor esforço
  - 7.7 Mecanismos de escalonamento e regulação
  - 7.8 Serviços integrados e serviços diferenciados
- 7.6 provendo garantias de QoS
  - 7.8 Serviços integrados e serviços diferenciados
  - 7.9 RSVP

# Fluxo Contínuo Multimídia Armazenada

Técnicas de *streaming* da camada de aplicação para extrair o máximo do serviço de melhor esforço

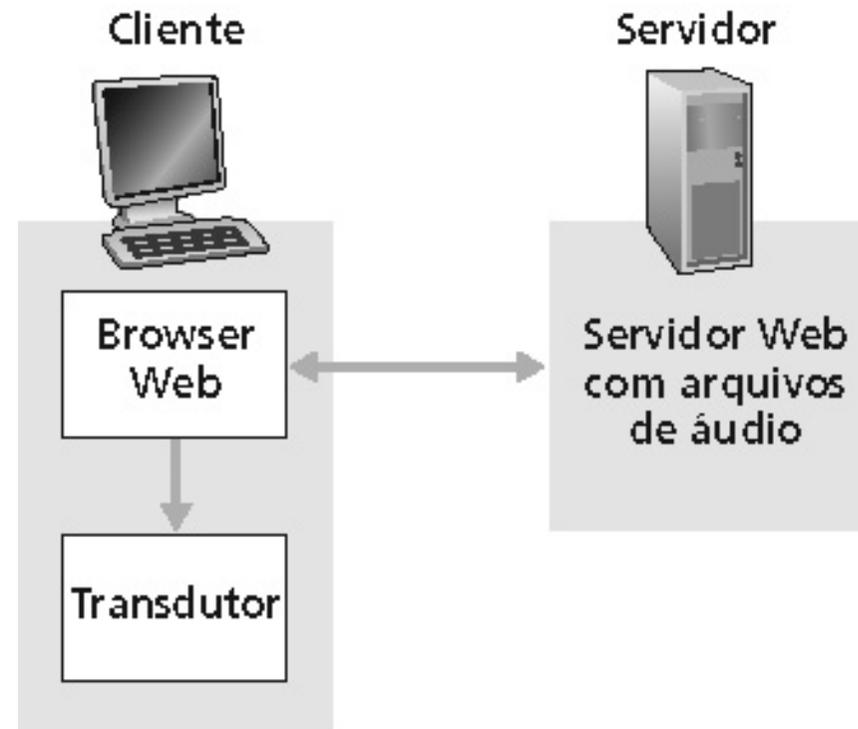
- armazenamento no lado do cliente
- uso do UDP ao invés do TCP
- múltiplas codificações da multimídia

Tocadores de Mídia (*transdutores*)

- descompressão
- Eliminação de variação de atraso (*jitter*)
- Tratamento de erros
- Interface gráfica do usuário com controles para interatividade

# Multimídia Internet: abordagem simplista

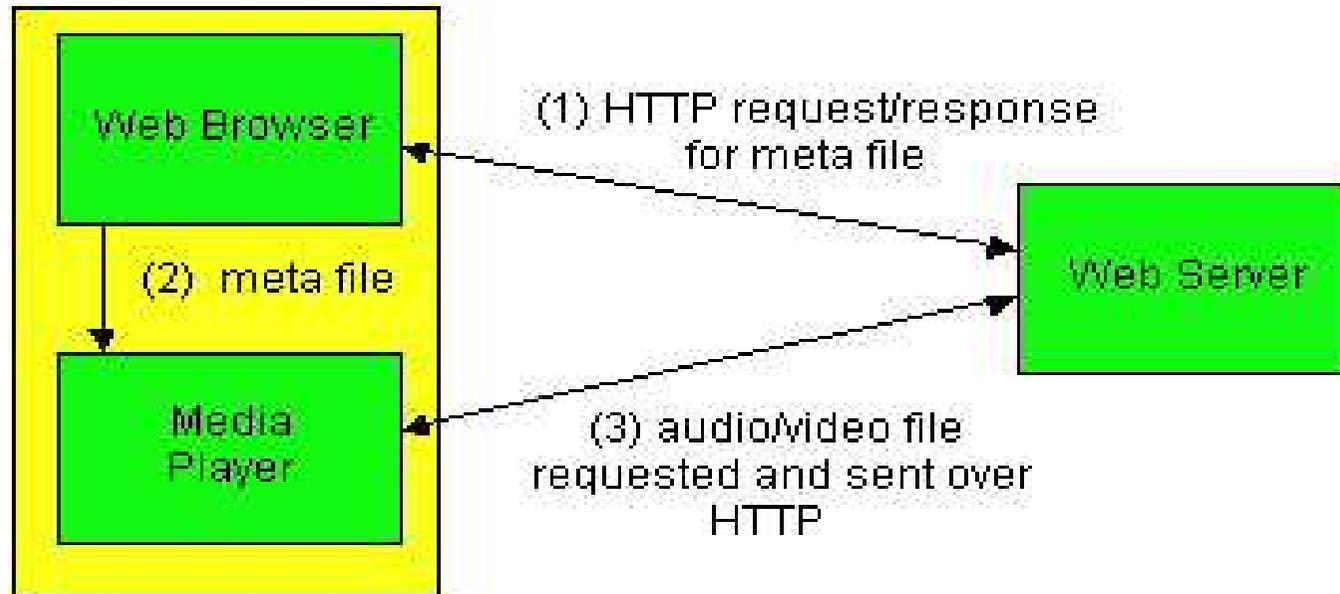
- áudio ou vídeo armazenado em arquivo
- arquivos transferidos como objetos HTTP
  - recebidos completamente pelo cliente
  - Depois repassado para o tocador de mídia (transdutor)



áudio, vídeo não são enviados como fluxo contínuo:

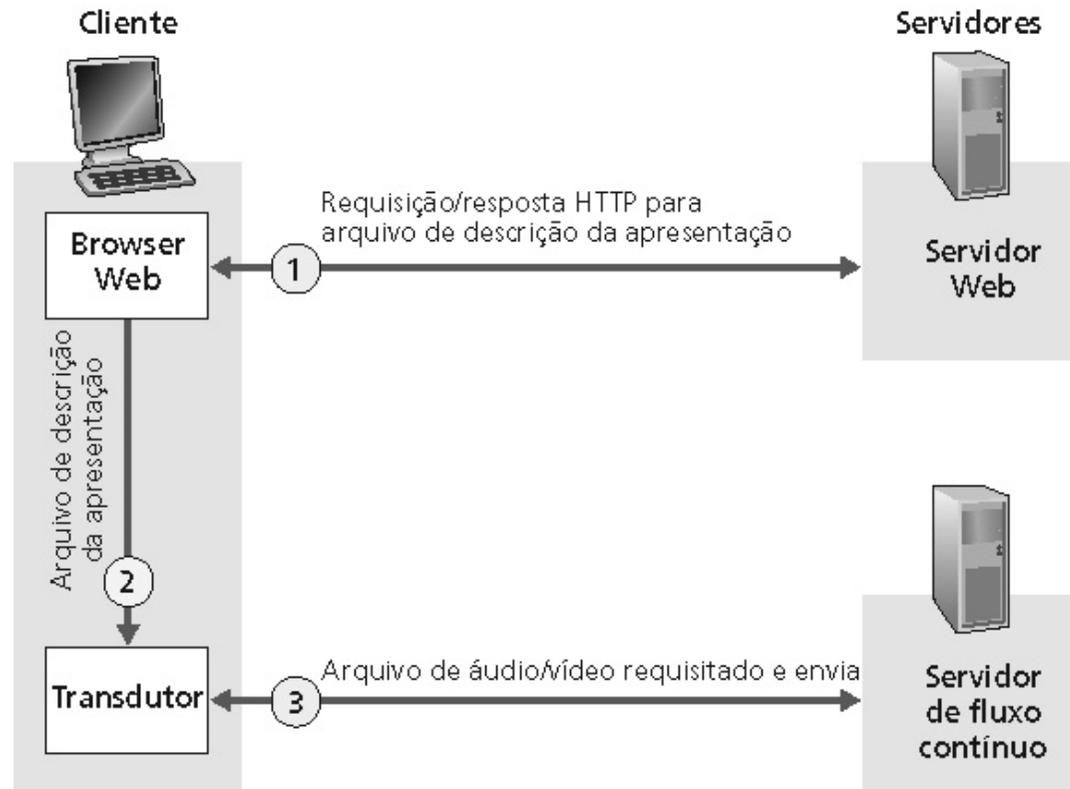
- ❑ não há cadeia de montagem (*pipelining*) o que resulta em longos atrasos até a reprodução!

# Multimídia Internet: abordagem com fluxos



- ❑ *browser* solicita **metarquivo**
- ❑ *browser* inicia o tocador (transdutor), passando o metarquivo
- ❑ Tocador (transdutor) contacta o servidor
- ❑ servidor cria o **fluxo** de áudio/vídeo até o tocador (transdutor)

# Fluxos a partir de um servidor de fluxo contínuo

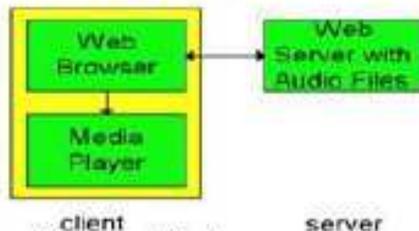


- Esta arquitetura permite o uso de protocolos não-HTTP entre o servidor e o reproduzidor de mídia (transdutor)
- Também pode usar UDP ao invés do TCP

# Fluxos Multimídia: Armazenamento pelo Cliente

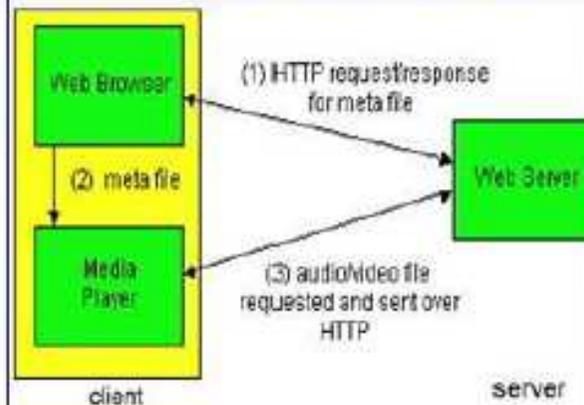
## Multimédia Internet: Abordagens

### Abordagem *Download*



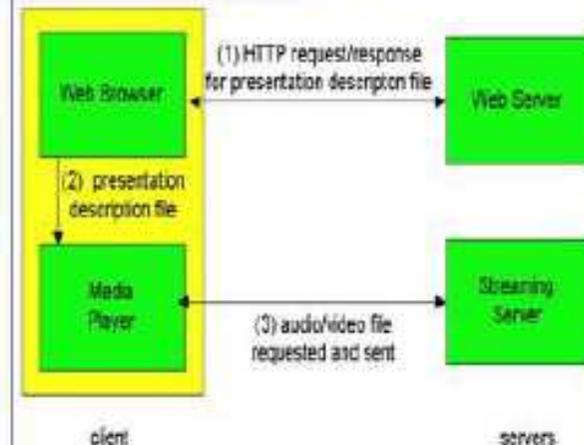
- **Áudio ou Vídeo armazenado em ficheiro**
- **Ficheiro transferido como objecto HTTP**
  - recebido 100% por cliente
  - depois segue para o *player*
- **Áudio, Vídeo não enviado como fluxo contínuo**
  - não há cadeia de montagem (*pipelining*)
  - longos atrasos até a reprodução

### Abordagem com fluxos



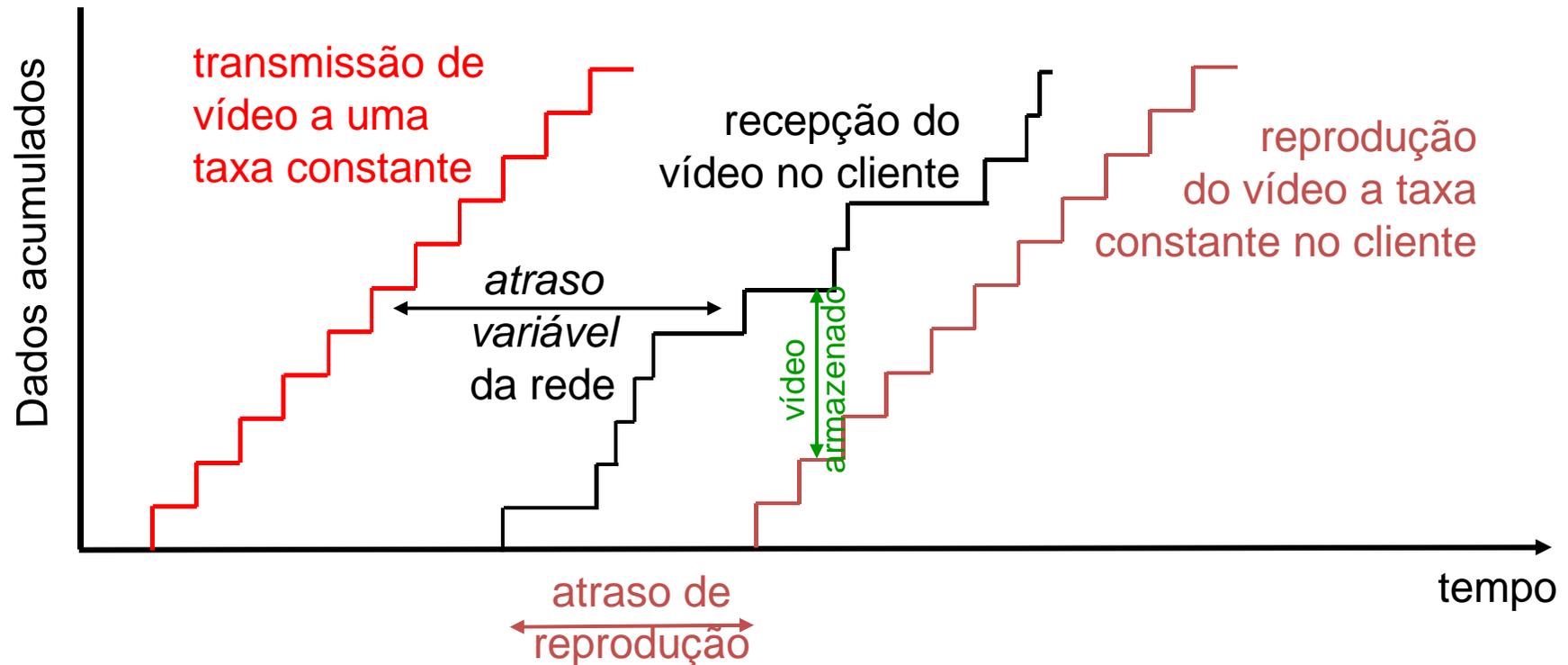
- *browser* solicita (GETs) **meta ficheiro**
- *browser* inicia o *player*, passando o meta ficheiro
- *player* contacta o servidor
- servidor cria o **fluxo** de áudio/vídeo até o *player*

### Servidor de fluxos



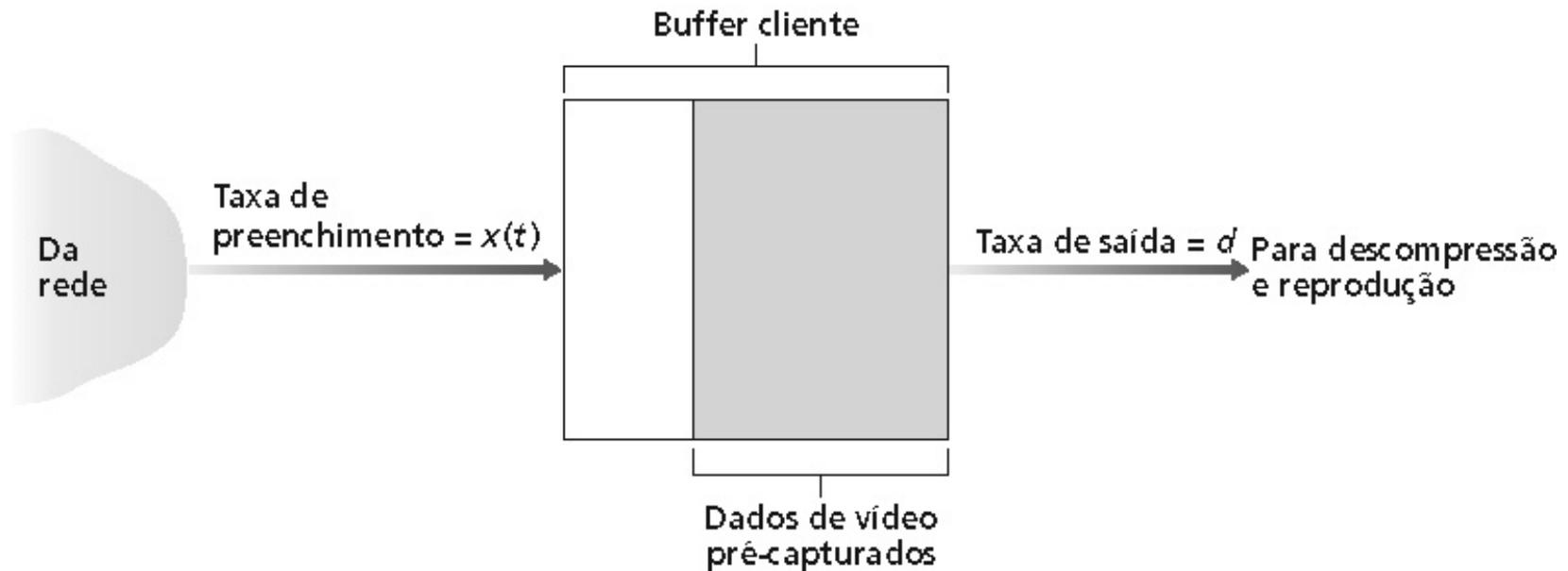
- Esta arquitectura permite o **uso de protocolos não-HTTP** entre o servidor e o reproduzidor de media
- Também pode usar UDP ao invés do TCP

# Fluxos Multimídia: Armazenamento pelo Cliente



- Armazenamento no lado do cliente, o atraso de reprodução compensa a variação do atraso (*jitter*) provocados pela rede

# Fluxos Multimídia: Armazenamento pelo Cliente



- Armazenamento no lado do cliente, o atraso de reprodução compensa a variação do atraso (*jitter*) provocados pela rede

# Fluxo Multimídia: UDP ou TCP?

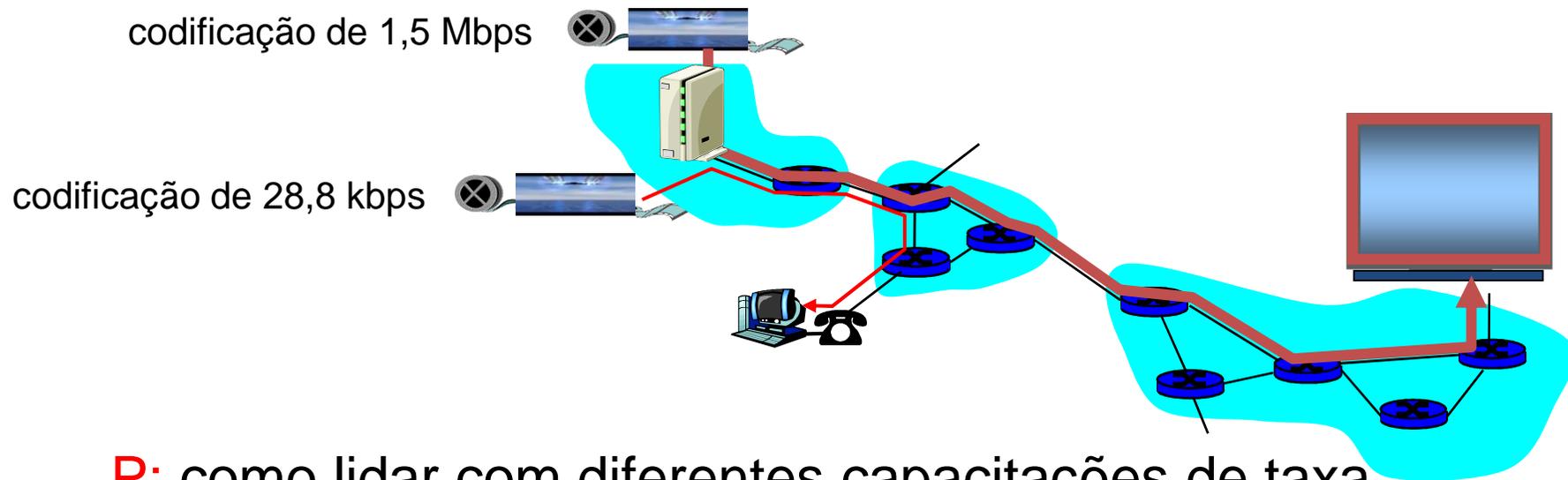
## UDP

- servidor envia a uma taxa adequada para o cliente (sem se importar com congestionamento da rede!)
  - freqüentemente taxa de transmissão = taxa de codificação = constante
  - portanto, taxa de enchimento = taxa constante – perda de pacotes
- pequeno atraso de reprodução (2-5 segundos) para compensar pela variação do atraso da rede
- recuperação de erros: se houver tempo

## TCP

- transmite na taxa máxima permitida pelo TCP
- taxa de enchimento flutua devido ao controle de congestionamento do TCP
- maior atraso para reprodução: taxa de entrega do TCP mais suave
- HTTP/TCP passam mais facilmente através de *firewalls*

# Fluxo Multimídia : taxa(s) do cliente



**P:** como lidar com diferentes capacidades de taxa de recepção do cliente?

- acesso discado de 28,8 kbps
- Ethernet de 100Mbps

**R:** servidor armazena, transmite múltiplas cópias do vídeo, codificadas em taxas diferentes

# Controle do Usuário de Fluxo Contínuo: RTSP

## HTTP

- Não tinha como alvo conteúdo multimídia
- Não possui comandos para avanço rápido, etc.

## RTSP: RFC 2326

- Protocolo cliente-servidor da camada de aplicação.
- O usuário pode controlar a apresentação: retorno, avanço rápido, pausa, retomada, reposicionamento, etc.

## O que ele não faz :

- Não define esquemas de compressão
- Não define como o áudio e vídeo são encapsulados para serem transmitido pela rede
- Não restringe como a mídia de fluxo contínuo é transportada; pode ser transportada por UDP ou TCP
- Não especifica como o apresentador da mídia armazena o áudio/vídeo

# RTSP: controle fora da faixa

## FTP usa um canal de controle “fora da faixa”:

- Um arquivo é transferido sobre uma conexão TCP.
- A informação de controle (mudanças de diretório, remoção de arquivo, renomeação de arquivo, etc.) é enviada numa conexão TCP à parte.
- Os canais “fora da faixa” e “dentro da faixa” utilizam diferentes números de portas.

## As mensagens RTSP também são enviadas fora da faixa:

- As mensagens de controle RTSP usam números de porta diferentes do fluxo da mídia, e são, portanto, enviadas fora da faixa
  - Porta 554
- O fluxo de mídia é considerado “dentro da faixa”.

# Exemplo RTSP

## Cenário:

- metarquivo enviado para o *browser* web
- *browser* inicia o tocador/transdutor
- Tocador/transdutor estabelece uma conexão de controle RTSP e uma conexão de dados para o servidor de mídia contínua

# 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>
```

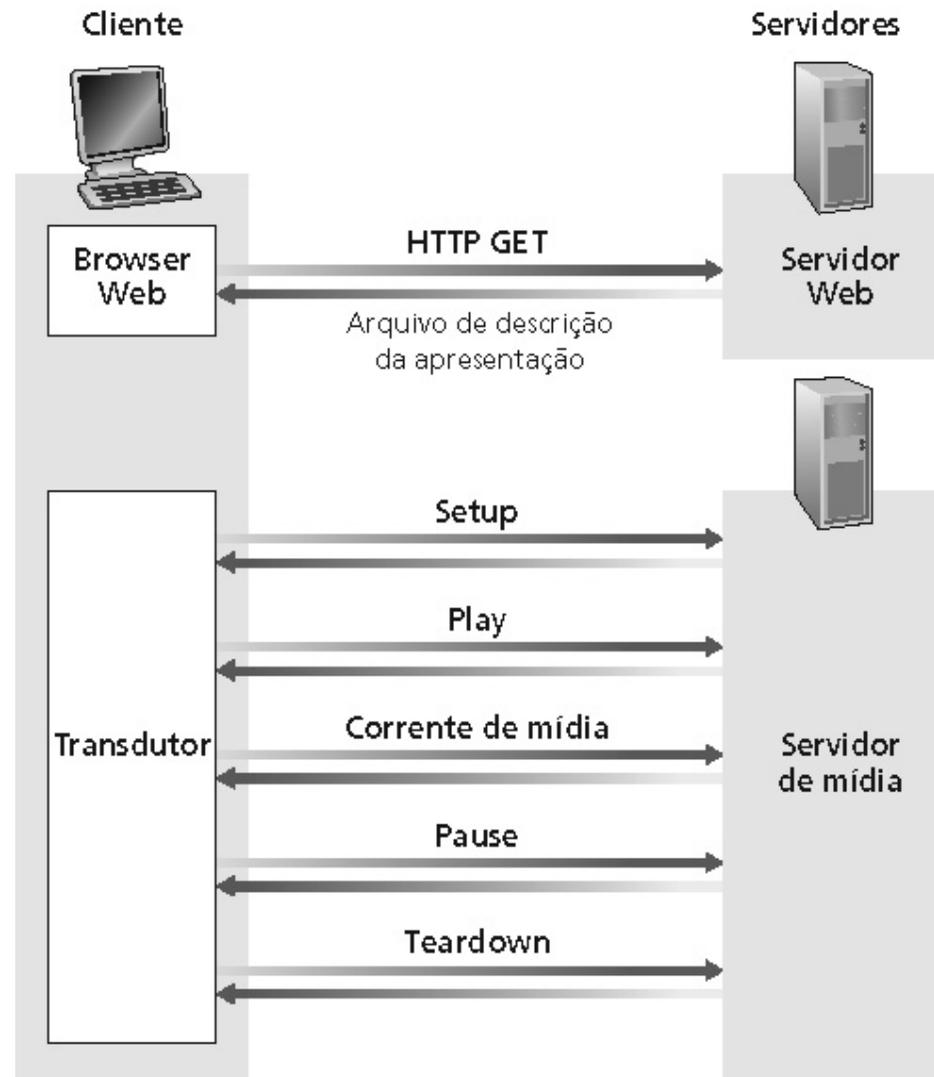
```
  <track type="video/jpeg"
```

```
    src="rtsp://video.example.com/twister/video">
```

```
</group>
```

```
</session>
```

# Operação do RTSP



# RTSP: exemplo de diálogo

C: SETUP rtsp://audio.example.com/twister/audio RTSP/1.0  
Cseq: 1  
Transport: rtp/udp; compression; port=3056; mode=PLAY

S: RTSP/1.0 200 OK  
Cseq: 1  
Session 4231

C: PLAY rtsp://audio.example.com/twister/audio.en/lofi RTSP/1.0  
Range: npt=0-  
Cseq: 2  
Session: 4231

S: RTSP/1.0 200 OK  
Cseq: 2  
Session 4231

C: PAUSE rtsp://audio.example.com/twister/audio.en/lofi RTSP/1.0  
Range: npt=37  
Cseq: 3  
Session: 4231

S: RTSP/1.0 200 OK  
Cseq: 3  
Session 4231

C: TEARDOWN rtsp://audio.example.com/twister/audio.en/lofi RTSP/1.0  
Cseq: 4  
Session: 4231

S: RTSP/1.0 200 OK  
Cseq: 4  
Session 4231

# Operação do RTSP



# Aplicações interativas em tempo real

- Telefonia PC-2-PC
  - Skype
- PC-2-telefone
  - Dialpad
  - Net2phone
  - Skype
- videoconferência com webcams
  - Skype
  - Polycom

Vamos agora examinar em detalhes um exemplo de telefonia Internet PC-2-PC

# Aplicações Interativas de Tempo Real

## Multimedia Interativa: Telefone Internet

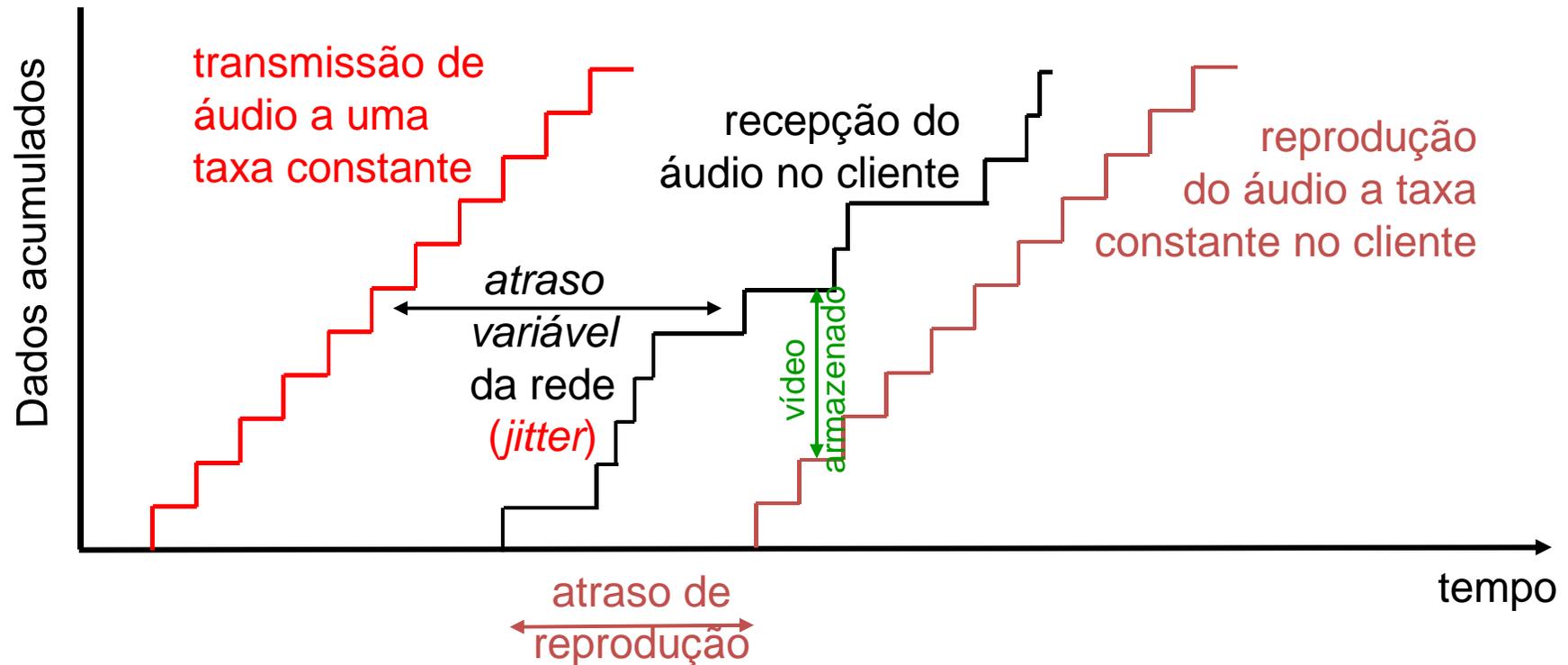
- Áudio do "speaker": alterna surtos de voz com períodos de silêncio
  - 64 kbps durante surto de voz
- Pacotes gerados apenas durante os surtos de voz
  - pedaços de 20 mseg a 8 Kbytes/seg: dados de 160 bytes
- Cabeçalho da camada de aplicação é adicionado a cada pedaço
- Pedaço+cabeçalho empacotado num segmento UDP
- Aplicação envia segmentos UDP no socket a cada 20 mseg durante um surto de voz.

- Telefone PC-2-PC
  - serviços de mensagens instantânea
- PC-2-telefone
  - teclado
  - Net2phone
- videoconferência com Webcams

# Telefone Internet: Perda e Atraso de Pacotes

- **perda pela rede:** datagrama IP perdido devido a congestionamento da rede (estouro do buffer do roteador)
- **perda por atraso:** o datagrama IP chega muito tarde para ser tocado no receptor
  - atrasos: processamento, enfileiramento na rede; atrasos do sistema terminal (transmissor, receptor)
  - atraso máximo tolerável típico: 400 ms
- tolerância a perdas: a depender da codificação da voz, as perdas podem ser encobertas, taxas de perdas de pacotes entre 1% e 10% podem ser toleradas.

# Variação do atraso (*jitter*)



- Considere o atraso fim a fim de dois pacotes consecutivos: a diferença pode ser maior ou menor do que 20 ms

# Telefone Internet

## – Perda de Pacotes e Atraso

- **Perda pela rede:** datagrama IP perdido devido a congestionamento da rede (estouro do buffer do router)
- **Perda por atraso:** o datagrama IP chega tarde para ser reproduzido no receptor
  - atrasos: processamento, filas de espera na rede; atrasos do sistema terminal (emissor, receptor)
  - atraso máximo tolerável típico: 400 ms
- **tolerância a perdas:** dependendo da codificação da voz, as perdas podem ser encobertas
  - taxas de perdas de pacotes entre 1% e 10% podem ser toleradas.

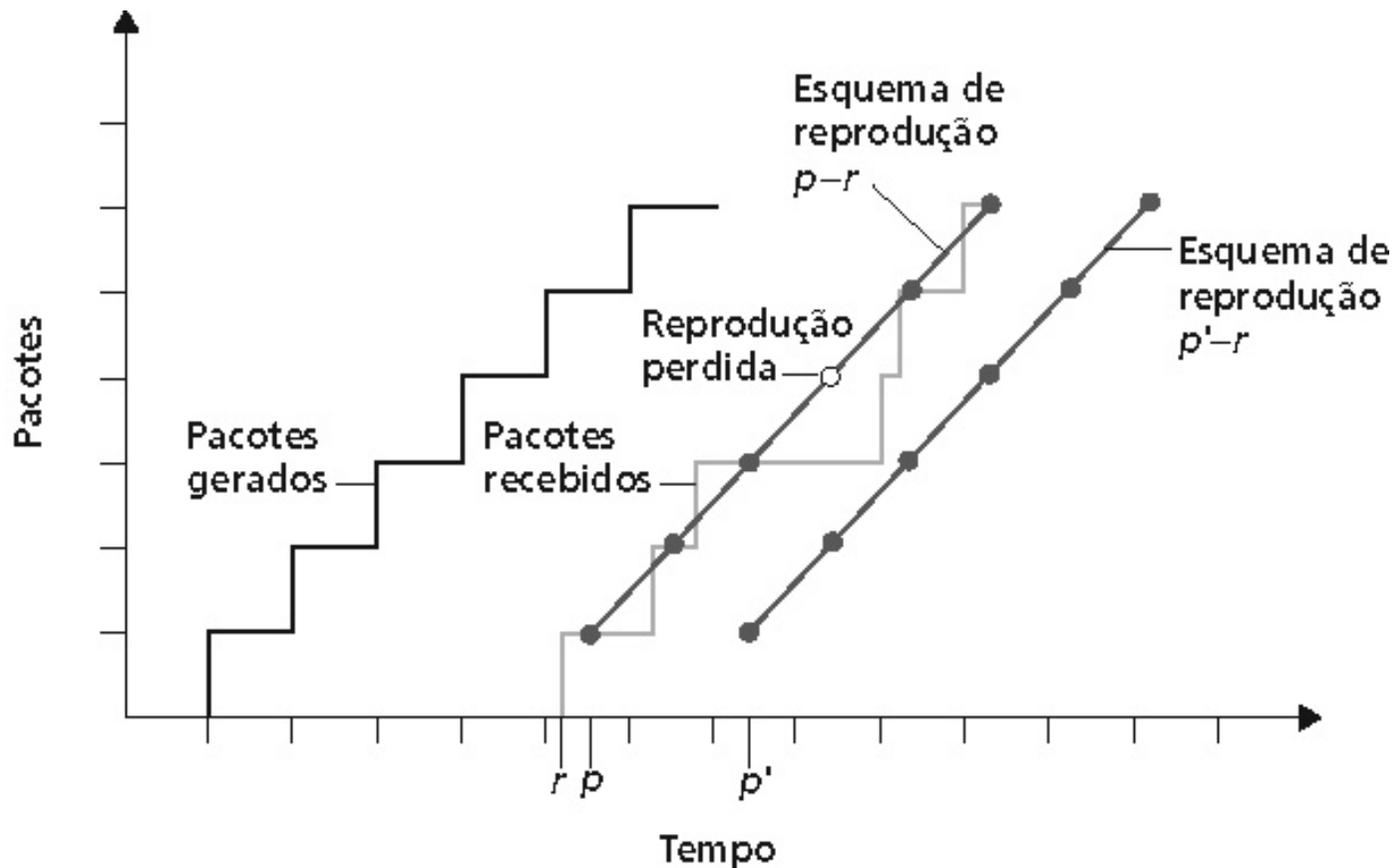
### Variação do atraso (jitter)



- Considerar o atraso fim a fim de dois pacotes consecutivos
  - a diferença pode ser maior ou menor do que 20 msec

# Atraso de reprodução fixo

- Transmissor gera pacotes a cada 20 ms durante um surto de voz.
- O primeiro pacote é recebido no instante  $r$
- A primeira reprodução é programada para iniciar no instante  $p$
- A segunda reprodução é programada para iniciar no instante  $p'$

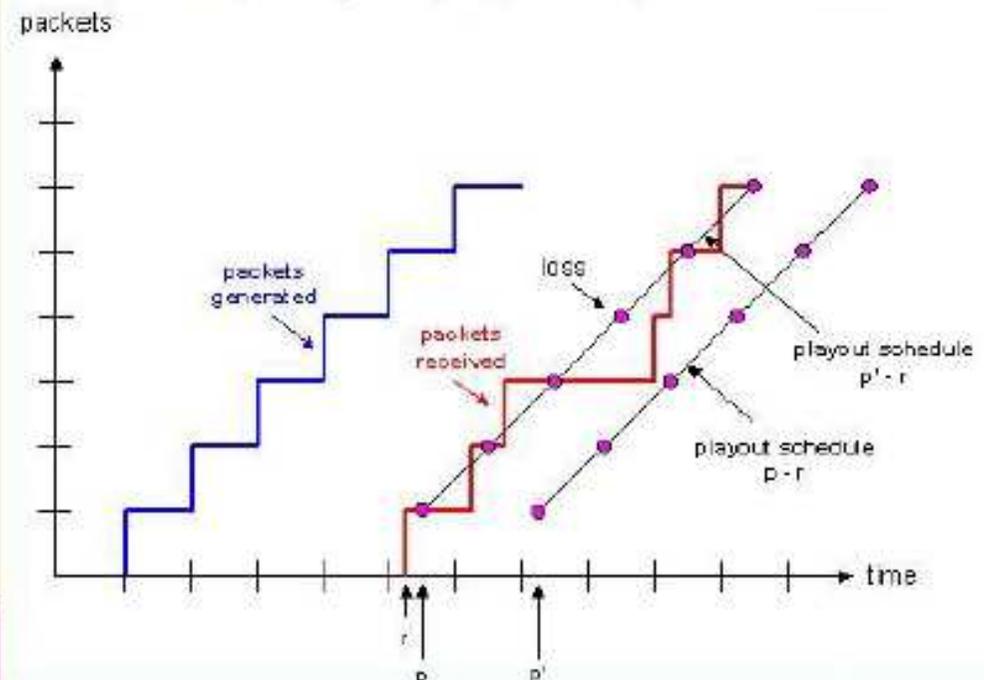


# Telefone Internet

## – Atraso de Apresentação Fixo

- O receptor tenta reproduzir cada pedaço exactamente  $q$  msecs após o pedaço ter sido produzido
  - Se o pedaço contiver um carimbo temporal  $t$ , o receptor reproduzirá o pedaço no instante  $t+q$ .
  - Se o pedaço chegar após o instante  $t+q$ , o receptor o descartará
- Compromissos para  $q$ :
  - $q$  longo: menos perda de pacotes
  - $q$  pequeno: melhor experiência interactiva

- Emissor gera pacotes a cada 20 mseg durante o surto de voz.
- O primeiro pacote é recebido no instante  $r$
- A primeira reprodução é programada para iniciar no instante  $p$ .
- A segunda reprodução é programada para iniciar no instante  $p'$



# Atraso de reprodução adaptativo

**Objectivo:** minimizar o atraso de reprodução, mantendo baixa a taxa de perdas

**Abordagem:** ajuste adaptativo atraso de reprodução:

- Estima o atraso da rede e ajusta o atraso de reprodução no início de cada surto de voz
- Períodos de silêncio são comprimidos e alongados.
- Os pedaços ainda são reproduzidos a cada 20 msec durante um surto de voz.

$t_i$  = carimbo de tempo do  $i$ -ésimo pacote

$r_i$  = instante em que o pacote  $i$  é recebido pelo receptor

$p_i$  = instante em que o pacote  $i$  é reproduzido no receptor

$r_i - t_i$  = atraso da rede para o  $i$ -ésimo pacote

$d_i$  = estimativa atraso médio da rede após o  $i$ -ésimo pacote

Estimativa dinâmica do atraso médio no receptor:

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

Versão filtrada do atraso do pacote

onde  $u$  é uma constante (ex.,  $u = 0,01$ ).

Estimava do desvio médio do atraso,  $v_i$ :

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

As estimativas  $d_i$  e  $v_i$  são calculadas para cada pacote recebido, mas são usados apenas no início de um surto de voz.

Para o primeiro pacote de um surto de voz, o tempo de apresentação é:

$$p_i = t_i + d_i + Kv_i$$

onde  $K$  é um constante positiva.

Os pacotes restantes em um surto de voz são reproduzidos periodicamente

# Reprodução adaptativa (3)

P: Como o receptor determina se um pacote é o primeiro de uma rajada de voz?

- Se nunca houvesse perdas, o receptor poderia simplesmente olhar as marcas de tempo sucessivas.
  - Diferença entre marcas sucessivas  $> 20$  ms, início da rajada de voz.
- Mas, dado que perdas são possíveis, o receptor deve olhar tanto para as marcas de tempo quanto para os números de seqüência.
  - Diferença entre marcas sucessivas  $> 20$  ms e números de seqüência sem falhas, início da rajada de voz.

# Recuperação de perda de pacotes (1)

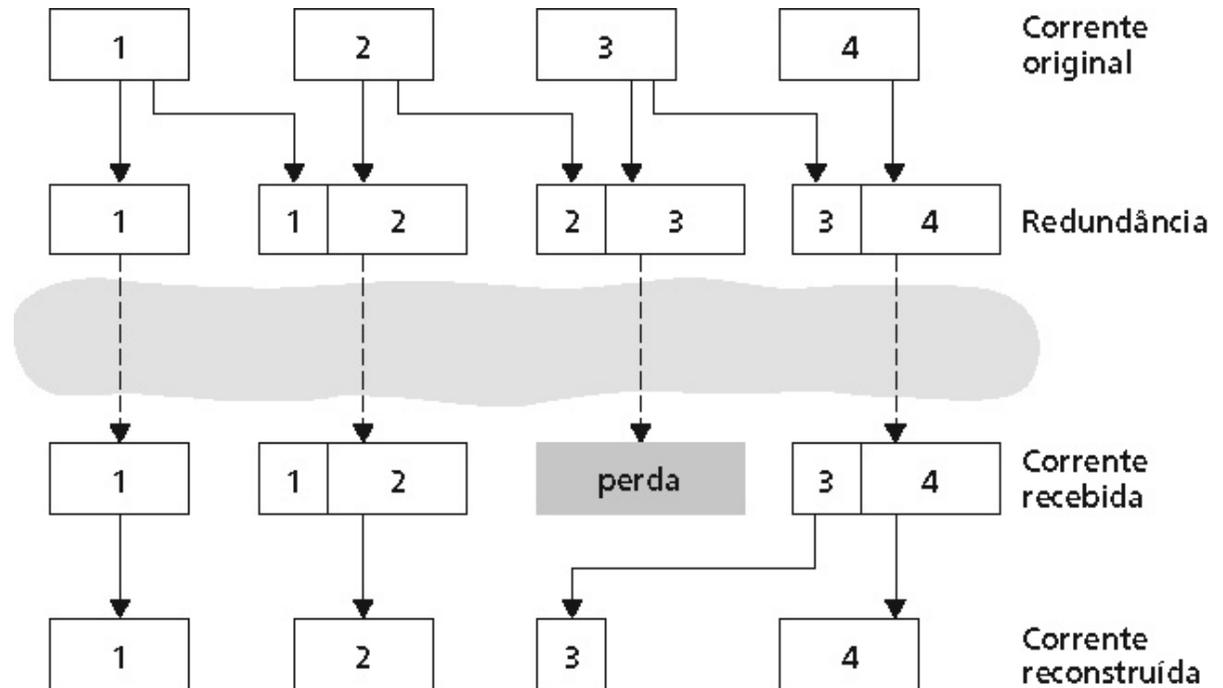
## forward error correction (FEC): esquema simples

- para cada grupo de  $n$  porções, criar uma porção redundante efetuando o OU-exclusivo das  $n$  porções originais
  - transmite  $n+1$  porções, aumentando a largura de banda por um fator de  $1/n$ .
  - pode reconstruir as  $n$  porções originais se houver no máximo uma porção perdida dentre as  $n+1$  porções.
- Atraso de reprodução deve ser fixado para o instante de recepção de todas as  $n+1$  porções
  - Compromissos:
    - aumento de  $n$ , menos desperdício de banda
    - aumento de  $n$ , atraso de reprodução mais longo
    - aumento de  $n$ , maior probabilidade de que 2 ou mais porções sejam perdidas

# Recuperação de perda de pacotes (2)

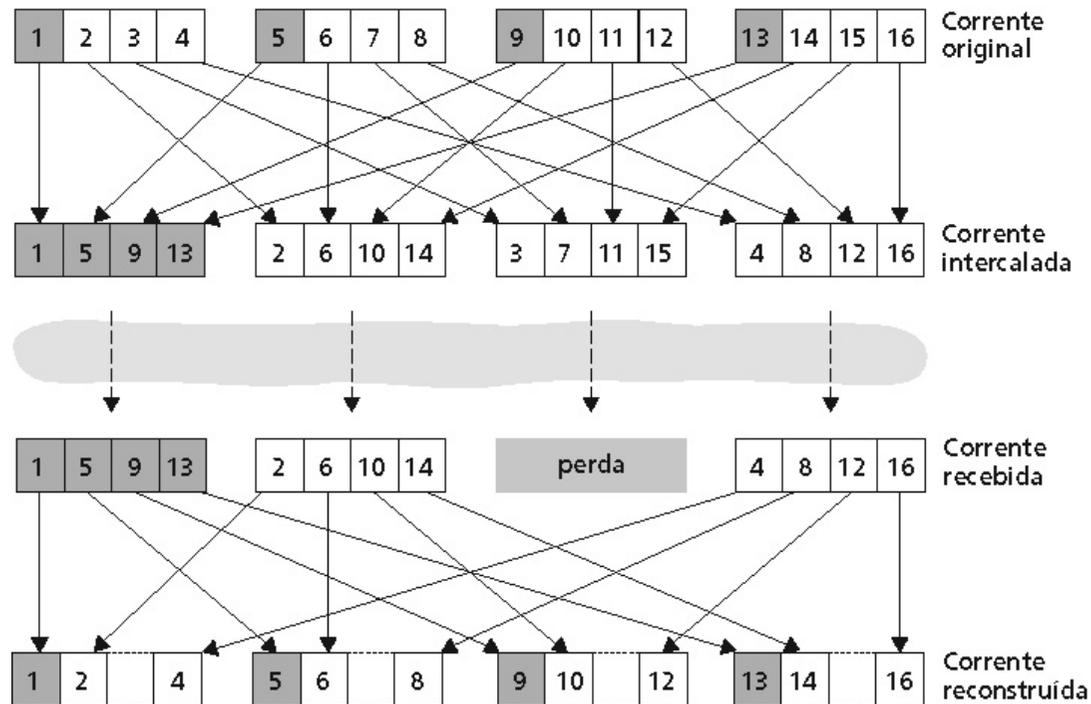
## 2o. Esquema de FEC

- transmissão de “carona” de um fluxo/corrente de menor qualidade
- envia fluxo de áudio de baixa resolução como informação redundante
- por exemplo, fluxo nominal PCM a 64 kbps e fluxo redundante GSM a 13 kbps.



- Sempre que houver uma perda não consecutiva, o receptor poderá recuperar a perda.
- Pode também adicionar a (n-1)-ésima e a (n-2)-ésima porção de baixa taxa de transmissão

# Recuperação de perda de pacotes (3)



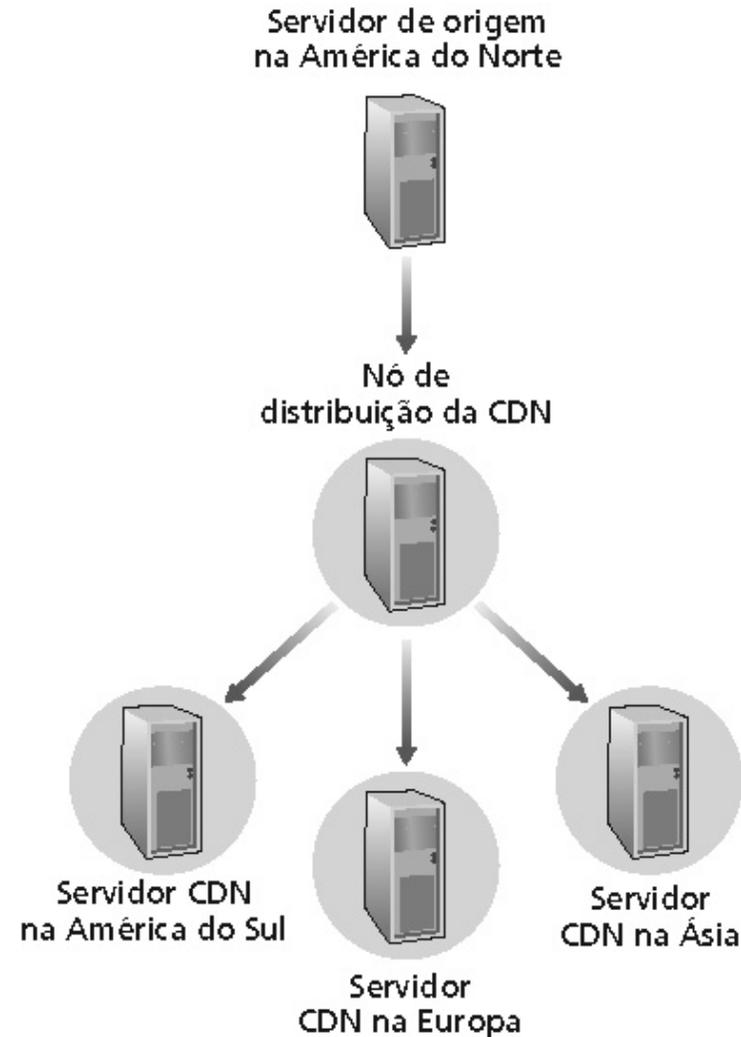
## Intercalação

- as porções são quebrados em unidades menores
- por exemplo, quatro unidades de 5 mseg por porção
- pacote agora contém pequenas unidades de porções diferentes
- se o pacote se perder, ainda temos muito de cada porção
- não tem sobrecarga de redundância
- mas aumenta o atraso de reprodução

# Redes de distribuição de conteúdo (CDNs - *Content distribution networks*)

## Replicação de conteúdo

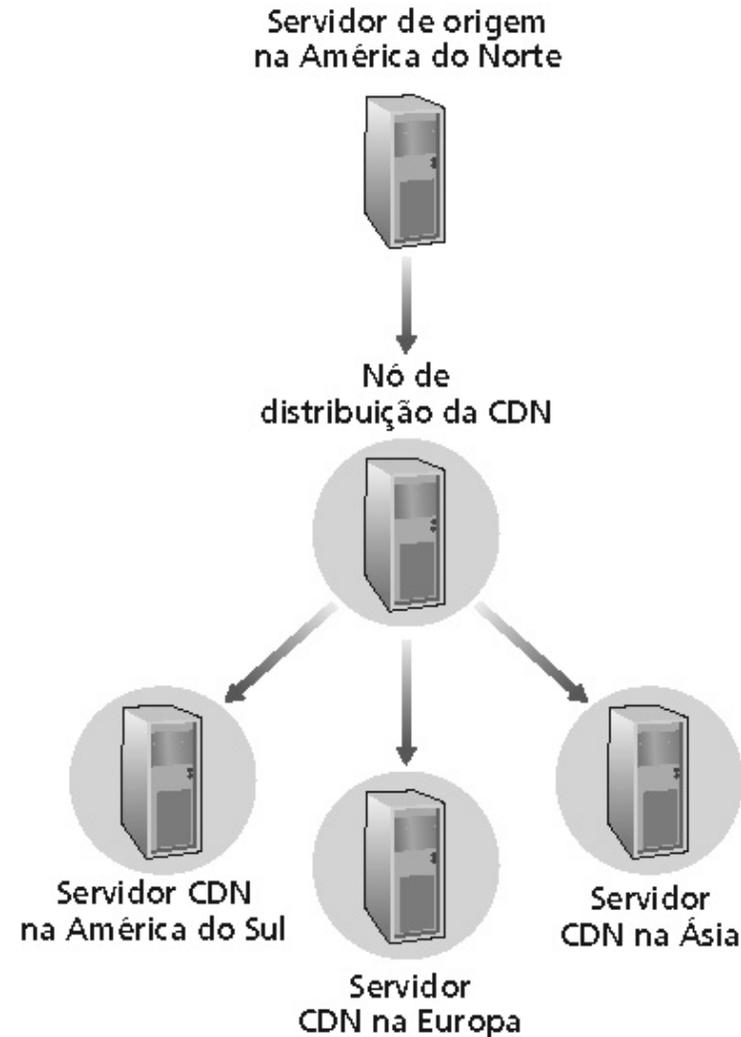
- Desafio transmitir fluxo de grandes arquivos (ex, vídeo) de um único servidor origem em tempo real
- Solução: replicar o conteúdo em centenas de servidores através da Internet
  - conteúdo carregado antecipadamente nos servidores CDN
  - Colocando o conteúdo “perto” do usuário evita impedimentos (perda, atraso) com o envio do conteúdo sobre caminhos longos.
  - servidor CDN tipicamente posicionado na borda da rede



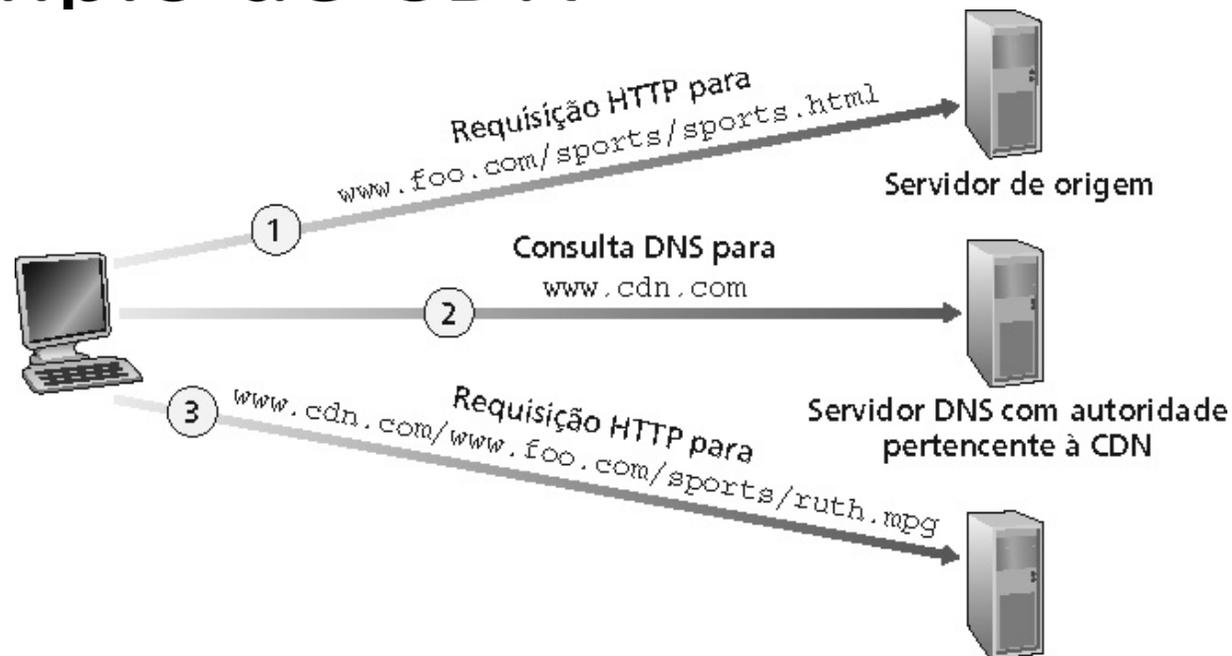
# Redes de distribuição de conteúdo (CDNs - *Content distribution networks*)

## Replicação de conteúdo

- Usuário de uma CDN (ex, Akamai) é o provedor de conteúdo (ex. CNN)
- A CDN replica o conteúdo do usuário em servidores CDN. Quando o provedor atualiza o conteúdo, a CDN atualiza os servidores



# Exemplo de CDN



## Servidor origem (www.foo.com)

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

## Empresa CDN (cdn.com)

- distribui arquivos gif
- usa o seu servidor DNS oficial para redirecionar os pedidos

# Mais sobre CDNs

## roteamento de pedidos

- A CDN cria um “mapa”, indicando as distâncias entre os ISPs folhas e os nós CDN
- quando a solicitação chega num servidor DNS oficial:
  - o servidor determina qual é o ISP de onde provém o pedido
  - usa o “mapa” para determinar qual o melhor servidor CDN
- nós CDN criam uma rede sobreposta na camada de aplicação

# Resumo: Multimídia Internet: truques

- **use UDP** para evitar o(s) (atrasos) do controle de congestionamento do TCP para tráfego sensível a tempo
- **atraso de reprodução adaptativo** para o lado do cliente: para compensar o atraso
- o lado do servidor **casa a largura de banda do fluxo** à largura de banda disponível no caminho cliente-ao-servidor
  - escolha entre taxas de fluxos pré-codificadas
  - taxa dinâmica de codificação do servidor
- recuperação de erros (acima do UDP)
  - FEC, intercalação, encobrimento de erros
  - retransmissões, se houver tempo
- CDN: traz o conteúdo mais para perto dos clientes

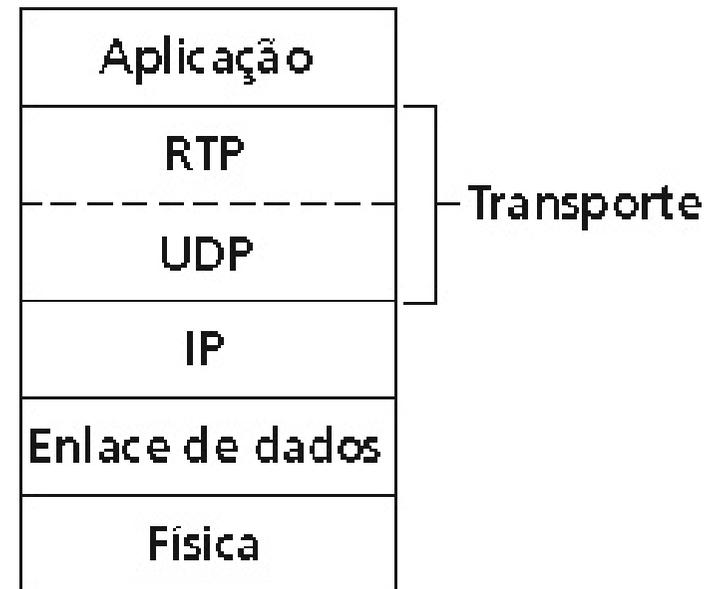
# Protocolo de Tempo Real (RTP)

- RTP = *Real Time Protocol*
- RTP especifica uma estrutura de pacote para pacotes que transportam dados de áudio e de vídeo
- RFC 3550
- Pacote RTP provê
  - Identificação do tipo da carga
  - Numeração da seqüência de pacotes
  - Marca de tempo
- RTP roda nos sistemas terminais.
- Pacotes RTP são encapsulados em segmentos UDP
- Interoperabilidade: Se duas aplicações de telefone Internet rodarem RTP então elas poderão trabalhar em conjunto

# RTP roda sobre UDP

Bibliotecas RTP provêm uma interface da camada de transporte que estende o UDP:

- números de portas, endereços IP
- verificação de erro através de segmentos
- identificação do tipo da carga
- numeração da seqüência de pacotes
- marca de tempo



# Exemplo RTP

- Considere o envio de voz codificada em PCM de 64 kbps sobre RTP.
- Aplicação coleta os dados codificados em porções, ex., a cada 20 ms = 160 bytes numa porção.
- A porção de áudio junto com o cabeçalho RTP formam um pacote RTP, que é encapsulado num segmento UDP.
- O cabeçalho RTP indica o tipo da codificação de áudio em cada pacote:
  - os transmissores podem mudar a codificação durante a conferência.
- O cabeçalho RTP também contém número de seqüência e marca de tempo.

# RTP e QoS

- RTP **não** provê nenhum mecanismo para garantir a entrega em tempo dos dados nem nenhuma outra garantia de qualidade de serviço.
- O encapsulamento RTP é visto apenas nos sistemas finais – não é visto por roteadores intermediários.
  - Roteadores provendo o serviço tradicional Internet de melhor esforço não fazem nenhum esforço adicional para garantir que os pacotes RTP cheguem ao destino em tempo.

# Cabeçalho RTP

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

## Cabeçalho RTP

**Tipo da carga útil (7 bits):** Usado para indicar o tipo de codificação que está sendo usada. Se o transmissor modificar a codificação no meio de uma conferência, o transmissor informará o receptor através do campo do tipo de carga útil.

- Tipo de carga 0: PCM lei  $\mu$ , 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, vídeo MPEG2

**Número de Seqüência (16 bits):** O número de seqüência é incrementado de um para cada pacote RTP enviado; pode ser usado para detectar a perda de pacotes e para restaurar a seqüência de pacotes.

# Cabeçalho RTP (2)

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

## Cabeçalho RTP

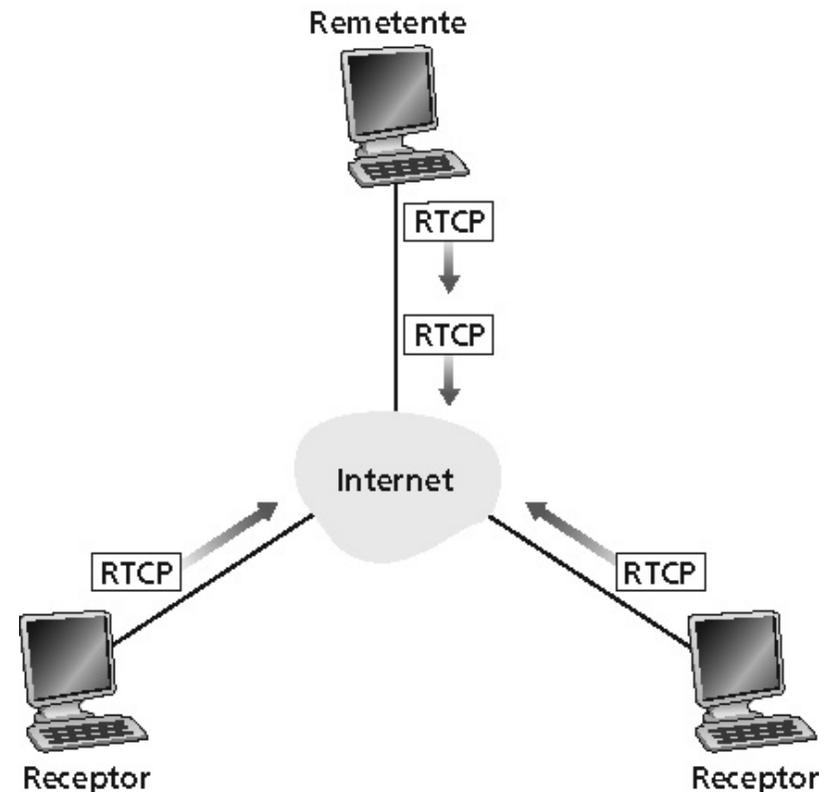
- **Campo de marca de tempo (32 bits):** Reflete o instante de amostragem do primeiro byte no pacote de dados RTP.
  - Para áudio o relógio de marca de tempo incrementa de um para cada período de amostragem (por exemplo, a cada 125  $\mu$ seg para um relógio de amostragem de 8kHz)
  - se a aplicação de áudio gerar porções de 160 amostras codificadas, então a marca de tempo aumenta de 160 para cada pacote RTP quando a fonte estiver ativa. O relógio de marca de tempo continua a aumentar a uma taxa constante mesmo quando a fonte estiver inativa.
- **Campo de identificador de sincronização da fonte (SSRC) (32 bits):** Identifica a fonte de um fluxo RTP. Cada fluxo numa sessão RTP deve possuir um SSRC distinto.

# Protocolo de Controle de Tempo Real (RTCP)

- *Real-Time Control Protocol*
- Trabalha em conjunto com o RTP.
- Cada participante numa sessão RTP periodicamente transmite pacotes de controle RTCP para todos os demais participantes.
- Cada pacote RTCP contém relatórios do transmissor e/ou receptor
  - relatam estatísticas úteis para as aplicações.
- Estas estatísticas incluem o número de pacotes enviados, o número de pacotes perdidos, *jitter* entre chegadas, etc.
- Esta realimentação de informação para as aplicações pode ser usada para controlar o desempenho
  - O transmissor pode modificar as suas taxas de transmissão baseadas na realimentação.

# RTCP - Continuação

- Para uma sessão RTP há tipicamente um único endereço multicast; todos os pacotes RTP e RTCP pertencentes à sessão usam o endereço multicast.
- Pacotes RTP e RTCP são diferenciados uns dos outros através do uso de números de portas distintos.
- Para limitar o tráfego, cada participante reduz o seu tráfego RTCP à medida que cresce o número de participantes da conferência.



# Pacotes RTCP

## Pacotes de relatório do receptor:

- Fração dos pacotes perdidos, último número de seqüência, *jitter* entre chegadas.

## Pacotes de relatório do remetente:

- SSRC do fluxo RTP, marca de tempo, número de pacotes e número de bytes enviados.

## Pacotes de descrição da fonte:

- Endereço de e-mail do remetente, nome do remetente, o SSRC do fluxo RTP associado.
- Os pacotes provêm um mapeamento entre o SSRC e o nome do usuário/hospedeiro.

# Sincronização de Fluxos

- O RTCP pode ser usado para sincronizar diferentes fluxos de mídia dentro de uma sessão RTP.
- Considere uma aplicação de videoconferência para a qual cada transmissor gera um fluxo RTP para vídeo e outro para áudio.
- As marcas de tempo nestes pacotes RTP estão vinculadas aos relógios de amostragem de vídeo e de áudio, e não estão vinculadas ao relógio de tempo real.
- Cada pacote de relatório do remetente contém, para o pacote gerado mais recentemente no fluxo RTP associado,
  - a marca de tempo do pacote RTP
  - e instante num relógio de tempo real em que o pacote foi criado.
- Os receptores pode usar esta associação para sincronizar a reprodução de áudio e de vídeo.

# Escalabilidade da Largura de Banda do RTCP

- O RTCP tenta limitar o seu tráfego a 5% da largura de banda da sessão.

## Exemplo

- Suponha que haja um transmissor enviando vídeo a uma taxa de 2 Mbps. Então o RTCP tenta limitar o seu tráfego a 100 kbps.
- O protocolo atribui 75% desta taxa, ou 75 kbps, para os receptores; e atribui os restantes 25% da taxa, ou 25 kbps, para o transmissor.
- Os 75 kbps alocados são compartilhados igualmente entre os receptores:
  - se houver  $R$  receptores, então cada receptor pode transmitir tráfego RTCP a uma taxa de  $75/R$  kbps
- Transmissor pode transmitir tráfego RTCP a uma taxa de 25 kbps.
- Um participante (um transmissor ou receptor) determina o período de transmissão dos pacotes RTCP através do cálculo dinâmico do tamanho médio de um pacote RTCP (ao longo de toda a sessão) e dividindo o tamanho médio do pacote RTCP pela sua taxa alocada.

# Objetivos de Projeto do RSVP

1. acomodar **receptores heterogêneos** (larguras de banda diferentes ao longo dos caminhos)
2. acomodar diferentes aplicações **com diferentes requisitos de recursos**
3. tornar o **multicast um serviço de primeira classe**, com adaptação para participação em grupo multicast
4. **aproveitamento do roteamento multicast/unicast existente**, com adaptação a mudanças nas rotas unicast e multicast
5. **sobrecarga do protocolo de controle** com crescimento linear (no pior caso) em função do número de receptores
6. **projeto modular** para tecnologias heterogêneas

# O que o RSVP não faz

- ❑ especifica como os recursos devem ser reservados
  - ele é um mecanismo para comunicar as necessidades
- ❑ determina as rotas seguidas pelos pacotes
  - este é a tarefa dos protocolos de roteamento
  - sinalização desvinculada do roteamento
- ❑ interação com o repasse de pacotes
  - separação dos planos de controle (sinalização) e dados (repasse)

# RSVP: visão geral da operação

- Transmissores e receptor aderem a um grupo multicast
  - Realizado fora do RSVP
  - Transmissores não precisam se unir ao grupo
- Sinalização do transmissor para a rede
  - *Mensagem de caminho*: torna a presença do transmissor conhecida dos roteadores
  - Desligamento do caminho: remove o estado do caminho do transmissor dos roteadores
- Sinalização dos receptores para a rede
  - *Mensagem de reserva*: reserva recursos dos transmissores para o receptor
  - Remoção (*teardown*) das reservas: remove as reservas do receptor
- Sinalização da rede para o sistema final
  - Erro de caminho
  - Erro de reserva

# Redes Multimídia: Resumo

## Princípios

- classificação das aplicações multimídia
- identificação das necessidades de serviços de redes das aplicações
- extraíndo o máximo do serviço atual de melhor esforço

## Protocolos e Arquiteturas

- protocolos específicos para o melhor esforço
- mecanismos para fornecimento de QoS
- arquiteturas para QoS
  - múltiplas classes de serviço
  - Garantias de QoS, controle de admissão