

Aplicações Multimídia e Protocolos de Streaming

Profa. Débora Christina Muchaluat Saade
debora@midia.com.uff.br

Aplicações Multimídia e Protocolos de Streaming

Redes de Computadores II

- ø **Aplicações Multimídia**
 - *Requisitos*
- ø **Limitações da Internet Atual**
- ø **Controle da Apresentação de Mídia Contínua Armazenada**
 - *Procotolo RTSP*
- ø **Usando o serviço de melhor esforço**
 - *Compensação da variação do retardo*
- ø **Protocolos de Streaming**
 - *Protocolos RTP e RTCP*
- ø **Referência:**
 - *Capítulo 6 (Kurose, Redes de Computadores e a Internet)*

Aplicações Multimídia

- ø **Aplicações multimídia (mídia contínua)**
 - ***Sensíveis ao retardo e variação do retardo (jitter)***
 - Pacotes que sofrem retardos de centenas de ms (telefonia IP) até poucos segundos (recepção de mídia armazenada) são inúteis
 - ***Tolerantes a perdas***
 - Perdas ocasionais causam pequenas perturbações na recepção de áudio e vídeo
- ø **Essas características diferem das aplicações tradicionais (mídia discreta)**

Aplicações Multimídia

ð Classificação das aplicações multimídia:

- *Transmissão de mídia contínua armazenada*
- *Transmissão de mídia contínua ao vivo*
- *Transmissão de mídia contínua interativa*

Aplicações Multimídia

δ Aplicações Multimídia com mídia contínua armazenada

- *Conteúdo foi pré-gravado e armazenado em um servidor*
- *Clientes solicitam arquivos de áudio e vídeo de servidores, recebem a informação pela rede e a apresentam*
- *Usuário pode controlar a operação*
 - similar a um VCR: pause, resume, fast forward, rewind, etc.
- *Fluxo contínuo:*
 - Clientes reproduzem parte do conteúdo ao mesmo tempo em que recebem o restante pela rede
- *Reprodução contínua:*
 - Assim que se inicia a reprodução da mídia, ela deve prosseguir de acordo com a temporização original da gravação
 - Restrições ao atraso na entrega dos dados
- *Retardo:*
 - Resposta considerada aceitável se o tempo a partir do pedido do cliente até o início da apresentação for de 1 a 10 segundos

Aplicações Multimídia

- **Aplicações Multimídia com mídia contínua transmitida ao vivo**
 - *tempo-real unidirecional*
 - *similar à difusão de rádio e TV convencional, mas a transferência de informação é feita pela Internet*
 - *Se armazenar o fluxo no cliente, pode pausar e retroceder*
 - *Muitos clientes recebem o mesmo conteúdo simultaneamente*
 - Distribuição eficiente precisa de comunicação multicast
 - **Retardo:**
 - Resposta considerada aceitável se o tempo a partir do pedido do cliente até o início da apresentação for de 1 a 10 segundos

Aplicações Multimídia

δ Aplicações Multimídia com mídia contínua interativa

- *Tempo-real interativo*
- *Conferência de áudio ou de vídeo*
- *Mais exigente nos requisitos de retardo e variação do retardo que o tempo-real unidirecional por causa da necessidade de interatividade em tempo-real*
- *Retardos:*
 - *Aúdio:*
 - *< 150 ms bom*
 - *de 150 a 400 ms aceitável*

Limitações da Internet Atual

- ø **Arquitetura Internet fornece serviço de melhor esforço**
- ø **Não há garantias sobre o retardo ou sobre a variação do retardo**
 - *Congestionamento na rede causa problema*
 - *na Internet pública todos os pacotes recebem tratamento igual*
 - *Pacotes contendo áudio e vídeo interativo de tempo-real permanecem nas filas, como todos os outros.*
- ø **Projeto de aplicações multimídia seria mais fácil se houvesse várias classes de serviço**
 - *Esforços vêm sendo desenvolvidos para prover serviços diferenciados com garantias de QoS – qualidade de serviço.*

Aproveitando ao máximo o “melhor esforço”

Para reduzir o impacto do serviço de melhor esforço da Internet, nós podemos:

- δ Usar UDP para evitar o TCP e sua fase de partida lenta...
- δ Armazenar o conteúdo no cliente e controlar a apresentação para remediar o jitter
- δ Acrescentar marcas de tempo nos pacotes para que o receptor saiba quando reproduzi-los.
- δ Adaptar o nível de compressão à taxa de transmissão disponível
- δ Nós podemos transmitir pacotes redundantes para atenuar os efeitos das perdas de pacotes.

➔ **Nós discutiremos todas essas técnicas**

Como a Internet deveria evoluir para dar o suporte necessário?

- ð **Aumentar a largura de banda**
 - *Junto com aumento do armazenamento intermediário na rede*
 - *Problema para aplicações que demandam muito (HDTV sob demanda)*

- ð **Modificar a infraestrutura de transmissão existente**

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

Redes de Computadores II

Filosofia de serviços Integrados:

- δ Mudar os protocolos da Internet de forma que as aplicações possam reservar uma banda de transmissão fim-a-fim
 - *Necessita de um novo protocolo que reserva banda de transmissão*
 - *Deve modificar as regras de escalonamento nos roteadores para poder honrar às reservas*
 - *Aplicação deve fornecer à rede uma descrição do seu tráfego e deve posteriormente respeitar esta descrição.*
- δ Exige um novo e complexo software nos hosts e nos roteadores

Filosofia de serviços Diferenciados

Exige menos mudanças na infraestrutura da Internet, embora forneça serviços de primeira e de segunda classe.

- δ Datagramas são marcados.
- δ Usuários pagam mais para enviar e receber pacotes de primeira classe.
- δ ISPs pagam mais aos provedores de backbone para enviar e receber pacotes de primeira classe.

Aúdio e Vídeio Armazenados

δ Mídia Contínua armazenada

- *Arquivos de Aúdio e de Vídeio são armazenados em servidores*
- *Usuários solicitam os arquivos por demanda.*
- *Aúdio/vídeio são apresentados, digamos, 10s após o pedido.*
- *Controle da apresentação é permitido.*

δ Executor da mídia (player)

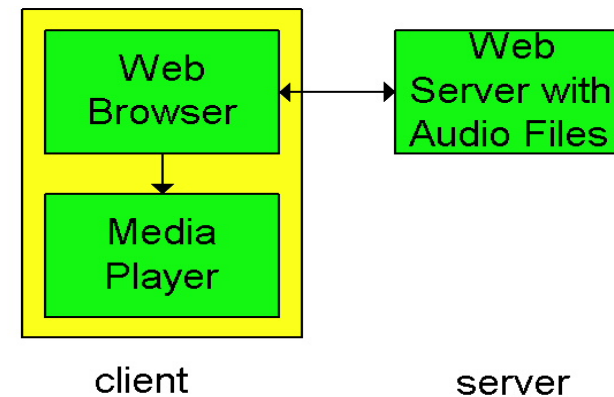
- *remove jitter (variação do retardo)*
- *Decodifica (descomprime) a mídia*
- *Realiza correção de erros*
- *Oferece interface gráfica para controle da apresentação*

δ Plug-ins podem ser usados para embutir o executor no browser web

- *3 abordagens para a implementação*

Acesso a Mídia Contínua a partir de servidores Web (1a. abordagem)

- ø **browser cliente solicita o arquivo com uma mensagem HTTP**
- ø **Servidor Web envia o arquivo na mensagem HTTP de resposta**
- ø **O cabeçalho “content-type” indica uma codificação apropriada para áudio e vídeo**
- ø **browser dispara o executor da mídia e passa o arquivo para ele**
- ø **executor apresenta o arquivo**



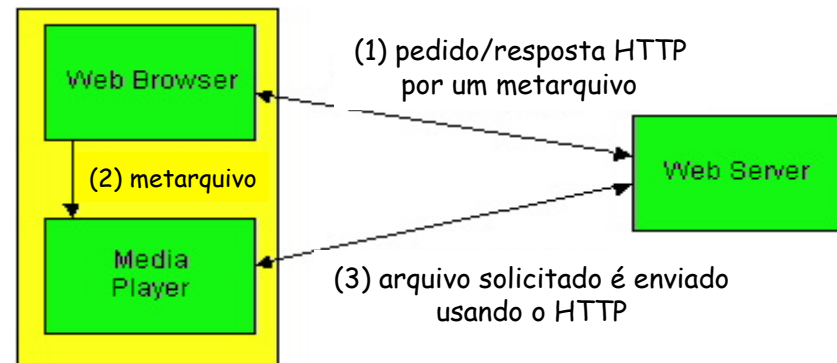
- **Maior problema: o executor de mídia interage com o servidor WEB através do Web browser que atua como intermediário.**

Acesso a Mídia Contínua a partir de servidores Web (2a. abordagem)

Redes de Computadores II

Alternativa: estabelecer conexão entre o servidor e o executor

- ø browser Web solicita e recebe um metarquivo (um arquivo descrevendo o objeto) ao invés de receber o próprio arquivo;
- ø O cabeçalho “Content-type” indica uma aplicação específica de áudio e vídeo
- ø Browser dispara o executor de mídia e passa o metarquivo para ele
- ø executor estabelece uma conexão TCP com o servidor e envia a ele a mensagem HTTP

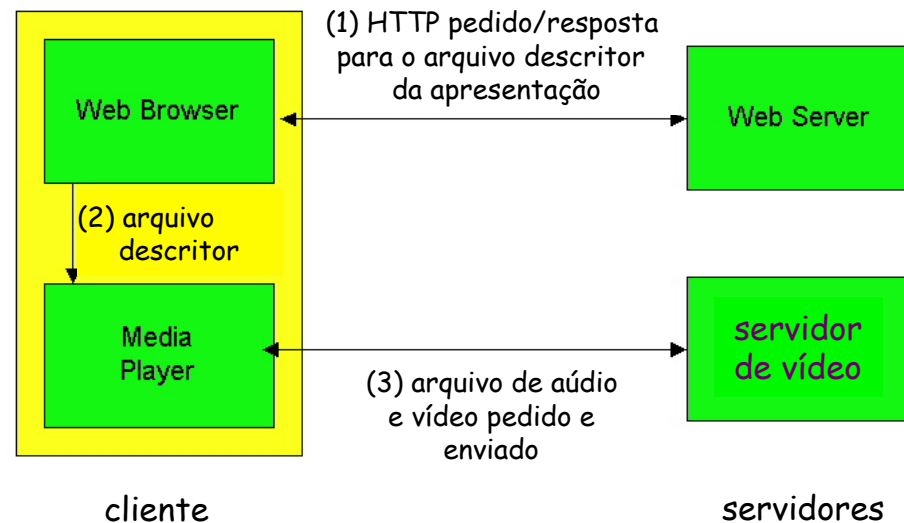


Algumas preocupações:

- ø O executor de mídia se comunica usando HTTP, que não foi projetado para suportar comandos de controle de apresentação
- ø Pode desejar enviar o áudio e o vídeo sobre UDP

Acesso a Mídia Contínua a partir de um servidor específico (3a. abordagem)

- Esta arquitetura permite o uso de outros protocolos (RTP, RTSP) (além do HTTP) entre o servidor e o executor de mídia (player)
- Pode usar também o UDP ao invés do TCP



Real Time Streaming Protocol: RTSP

RTSP: RFC 2326

- ø Protocolo de aplicação do tipo cliente-servidor.
- ø Permite ao usuário controlar apresentações de mídia contínua: voltar ao início, avançar, pausa, continuar, seleção de trilha, etc...

O que ele não faz:

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

Exemplo de uso: RealNetworks

- ø Servidor e cliente usam RTSP para enviar informações de controle de um para o outro

RTSP: controle fora da banda (out-of-band)

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

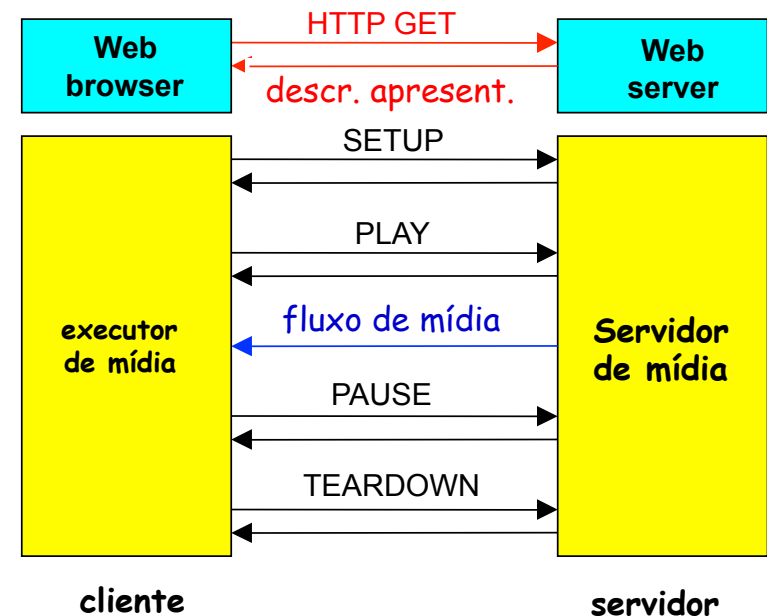
- ø Um arquivo é transferido sobre uma conexão TCP.
- ø Informação de controle (mudanças de diretório, remoção de arquivos, trocas de nomes, etc.) é enviada sobre uma conexão TCP separada.
- ø Os canais “dentro-da-banda” e “fora-da-banda” usam diferentes números de portas.

Mensagens RTSP também são enviadas “fora-da-banda”:

- ø As mensagens de controle RTSP usam diferentes números de portas em relação ao fluxo de dados de mídia contínua e, portanto, são enviadas “fora-da-banda”.
 - *RTSP usa porta 544 do TCP ou UDP*
- ø O fluxo de dados de mídia contínua, cuja estrutura de pacotes não é definida pelo RTSP, é considerada “dentro-da-banda”.

Iniciação do RTSP e controles de entrega

- ð Cliente obtém uma descrição da apresentação multimídia, que pode consistir de vários fluxos de dados (através do HTTP).
- ð O browser chama o executor de mídia (aplicação auxiliar) com base no tipo de conteúdo da descrição da apresentação.
- ð A descrição da apresentação inclui referências aos fluxos de mídia (URLs) usando o esquema “rtsp://...”
 1. *executor envia o comando RTSP SETUP; servidor envia a resposta RTSP SETUP.*
 2. *executor envia o comando RTSP PLAY; servidor envia a resposta RTSP PLAY.*
 3. *O servidor de mídia descarrega o fluxo de mídia.*
 4. *executor envia o comando RTSP PAUSE; o servidor envia a resposta RTSP PAUSE.*
 5. *executor envia o comando RTSP TEARDOWN; servidor envia a resposta RTSP TEARDOWN.*



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>
```

Sessão RTSP

- ø Cada sessão RTSP tem um identificador de sessão, que é escolhido pelo servidor.
- ø O cliente inicia a sessão com o comando **SETUP**, e o servidor responde ao comando com um identificador.
- ø O cliente repete o identificador de sessão em cada comando, até que o cliente encerra a sessão com o comando **TEARDOWN**.
- ø O número de porta do RTSP é 544.
- ø RTSP pode ser usado sobre UDP ou TCP. Cada mensagem RTSP pode ser enviada numa conexão TCP separada.

RTSP: exemplo de mensagens

**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