

# Capítulo 7 Multimedia en Redes de Computadores

Material basado en el Texto:  
Computer Networking: A Top Down Approach  
Featuring the Internet,  
3rd edition.  
Jim Kurose, Keith Ross  
Addison-Wesley, July 2004.

# Multimedia, Quality of Service: Qué es esto?

Aplicaciones Multimedia:  
audio y vídeo en la red  
entre otros ..  
("medio continuo")



## QoS

La red provee a las aplicaciones con *nivel de desempeño requerido para su funcionamiento.*

# Capítulo 7: Objetivos

## Principios

- Clasificar aplicaciones multimedia
- Identificar los servicios de la red requeridos por las aplicaciones
- Obtener lo mejor del servicio de “mejor esfuerzo”
- Mecanismos para proveer QoS

## Protocolos y arquitecturas

- Protocolos específicos para best-effort
- Mecanismos para proveer QoS
- Arquitecturas para QoS

# Capítulo 7: Contenidos

- 7.1 Aplicaciones Multimedia en Red
- 7.2 Streaming de Audio y video almacenado
- 7.3 Real-time Multimedia: Estudio de telefonía en Internet
- 7.4 protocolos para aplicaciones Interactivas de Tiempo Real
  - RTP, RTCP, SIP
- 7.5 Distribución de Multimedia: Redes de distribución de Contenidos
- 7.6 Más allá de Best Effort
- 7.7 Mecanismos de itineración y políticas
- 7.8 Servicios Integrados y Servicios Diferenciados
- 7.9 RSVP

# Aplicaciones Multimedia (MM) en red

## Clases de aplicaciones MM:

- 1) Streaming de audio y vídeo almacenado Ej. VoD
- 2) Streaming de audio y video en vivo Ej. Radio USM
- 3) Audio y vídeo en tiempo real e interactivo Ej. Skype

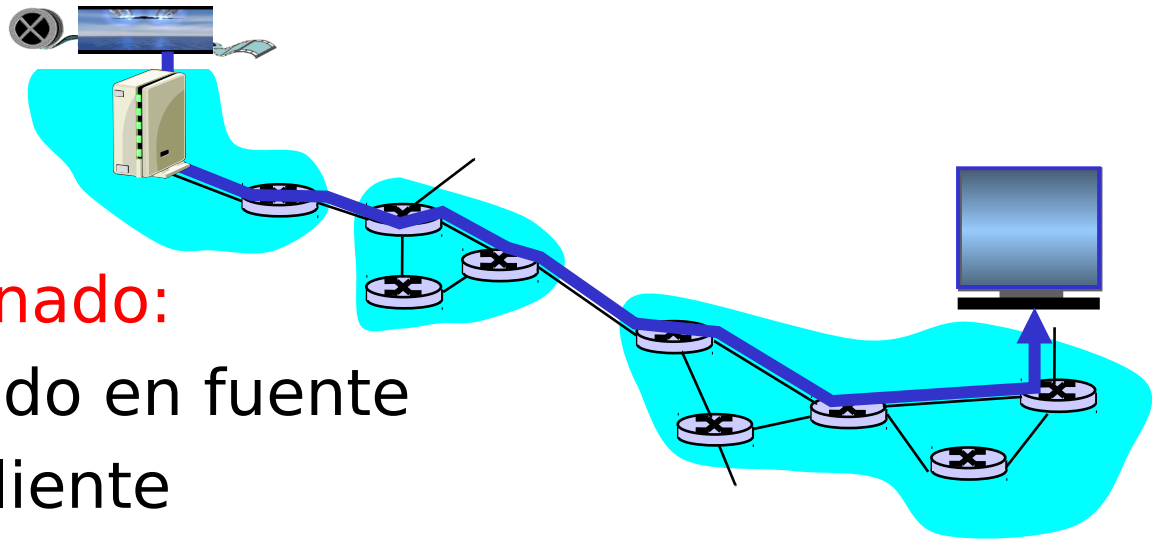
**Jitter** es la variabilidad del retardo de paquetes dentro del mismo flujo de paquetes

## Características

### Fundamentales:

- Típicamente **sensibilidad a retardo**
  - Retardo extremo-a-extremo
  - Variaciones de retardo (delay jitter)
- Pero **tolerante a pérdidas**: pérdidas no frecuentes causan distorsiones menores
- Antítesis de datos, los cuales son intolerante a pérdida pero tolerante a retardo.

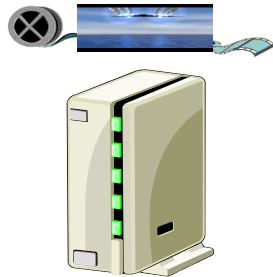
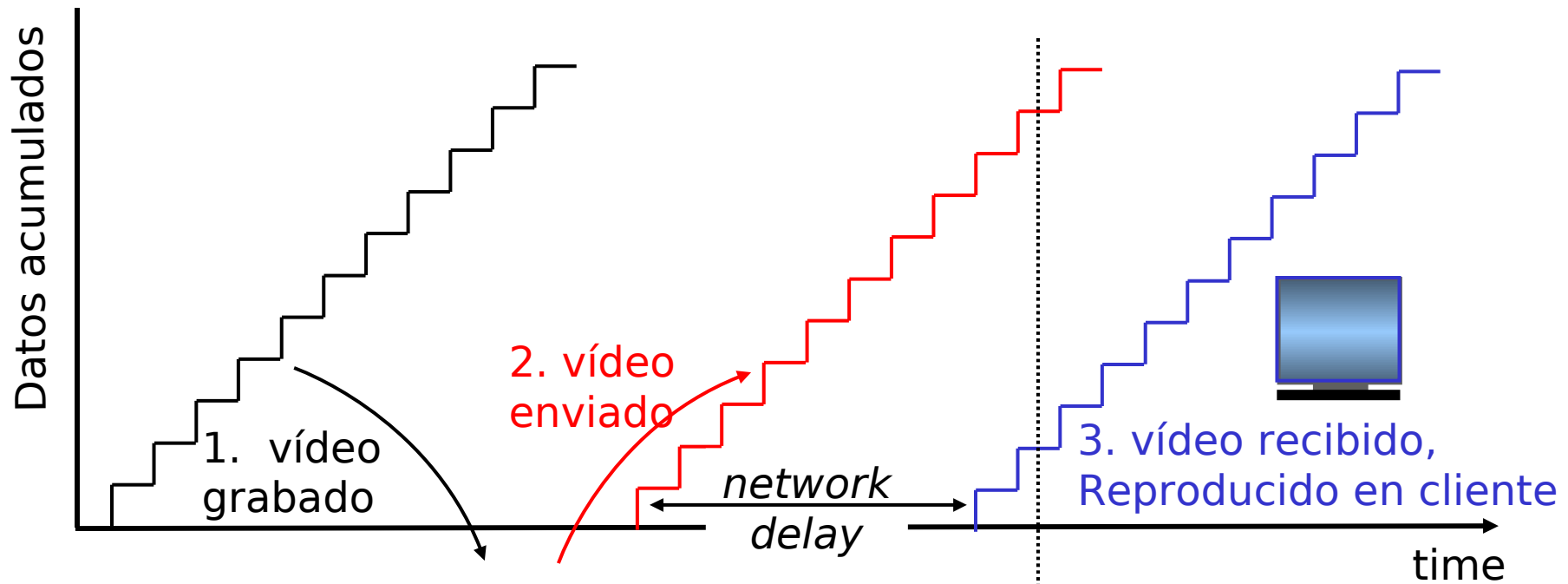
# Streaming de Multimedia almacenada



## Streaming almacenado:

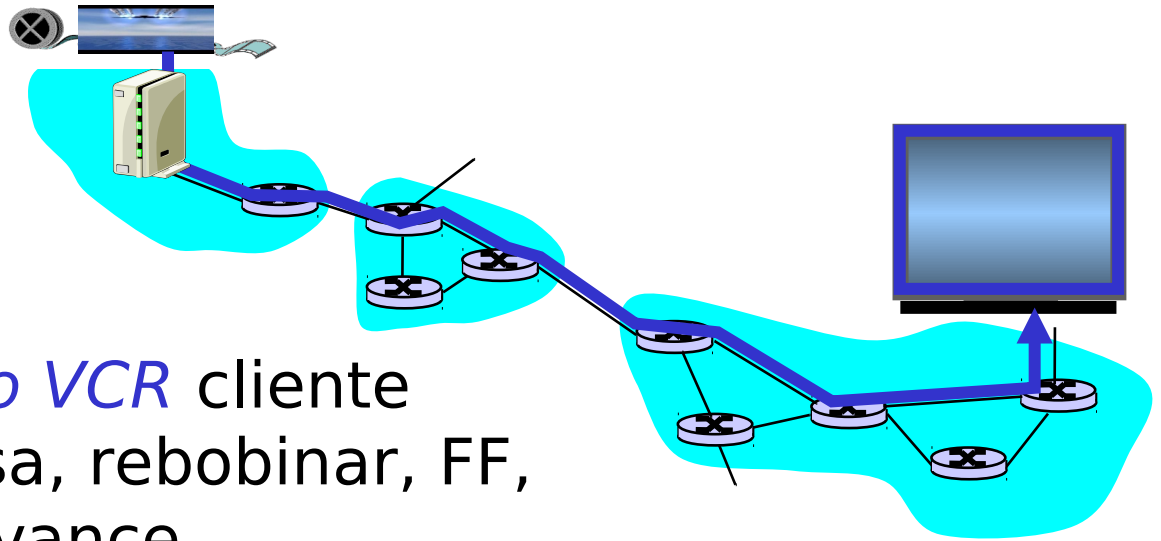
- ▣ Medio almacenado en fuente
- ▣ Transmitido al cliente
- ▣ streaming: reproducción en cliente comienza *antes que* todos los datos han llegado
- ▣ Restricción de tiempo para datos por ser transmitidos: llegar a tiempo para reproducción

# Streaming de Multimedia almacenada: ¿Qué es esto?



streaming: en este tiempo se reproduce la parte inicial del vídeo mientras el servidor aún envía lo siguiente

# Streaming de Multimedia almacenada: Interactividad



- *Funcionalidad tipo VCR* cliente puede hacer pausa, rebobinar, FF, mover barra de avance
  - 10 sec retardo inicial es OK
  - 1-2 sec hasta comando actúa, OK
  
- Restricciones de tiempo para datos por ser transmitidos: a tiempo para reproducción



# Streaming de Multimedia en vivo

## Ejemplos:

- Programa de conversación en radio en Internet
- Evento deportivo en vivo

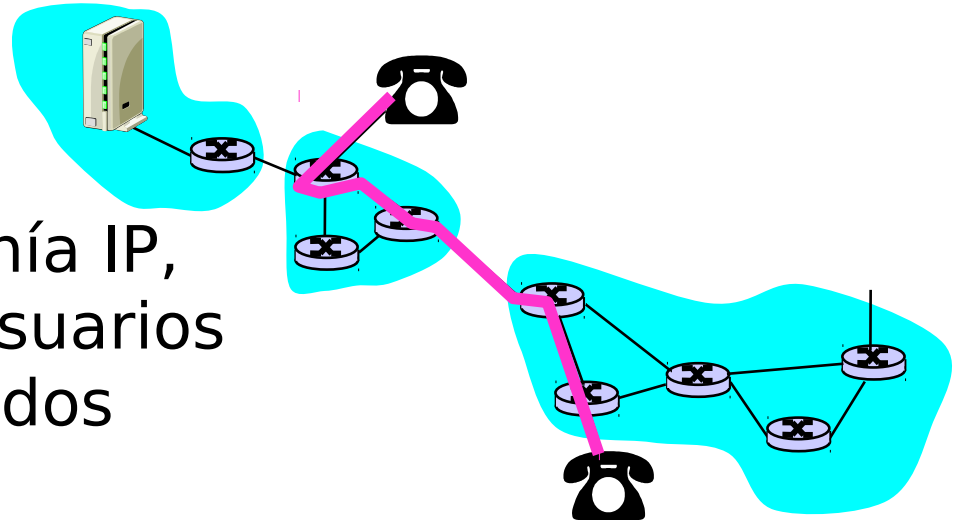
## Streaming

- Buffer de reproducción
- Reproducción puede retrasarse decenas de sec después de transmisión
- Aún tenemos restricciones de tiempo

## Interactividad

- fast forward es imposible
- Rewind y pause son posibles!
- “Time-shifting”

# Multimedia Interactiva, de tiempo real



- ▣ **Aplicaciones:** Telefonía IP, vídeo conferencia, usuarios interactivos distribuidos
- ▣ **Requerimiento de retardo extremo a extremo:**
  - ▣ audio: < 150 msec bueno, < 400 msec OK
    - Incluye retardos capa aplicación (paquetización) y red
    - Retardos mayores notorios, impide interactividad
- ▣ **Iniciación de sesión**
  - ▣ ¿Cómo avisa el llamado su IP, puerto, algoritmo de codificación?

# Multimedia en la Internet de hoy

**TCP/UDP/IP:** “servicio best-effort”

- ▮ *no* hay garantías de retardo ni pérdidas



Pero decimos que aplicaciones multimedia requieren QoS y niveles de desempeño para ser útiles!



Aplicaciones multimedia actuales usan técnicas en capa aplicación para mitigar (lo mejor posible) efectos de retardo y pérdidas

# ¿Cómo debería evolucionar Internet para mejorar soporte multimedia?

## Filosofía de servicios integrados:

- ▣ Cambios fundamentales en Internet, así aplicaciones pueden reservar ancho de banda extremo a extremo
- ▣ Requiere nuevo y complejo software en routers y hosts

## “Dejar-hacer, dejar-pasar”

- ▣ no mayores cambios
- ▣ Más BW cuando se necesite
- ▣ Redes de Distribución de contenidos (CDN), multicast a nivel aplicación

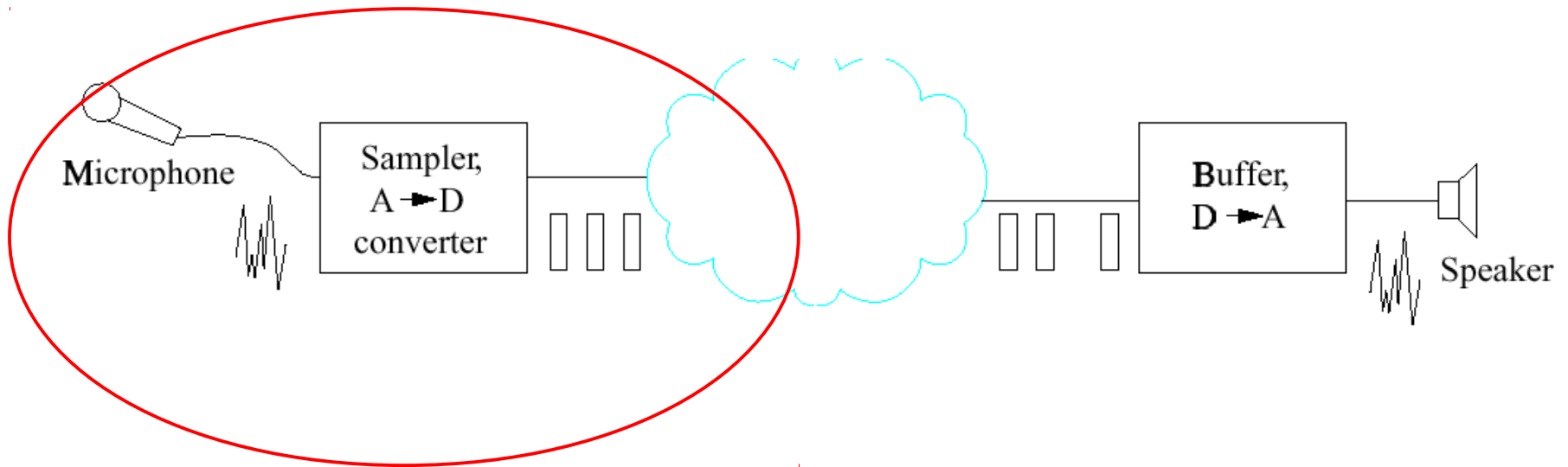
## Filosofía de Servicios diferenciados:

- ▣ Menores cambios a Internet, pero proveer servicio de 1ª y 2ª clase. Cobro de BW según clase.



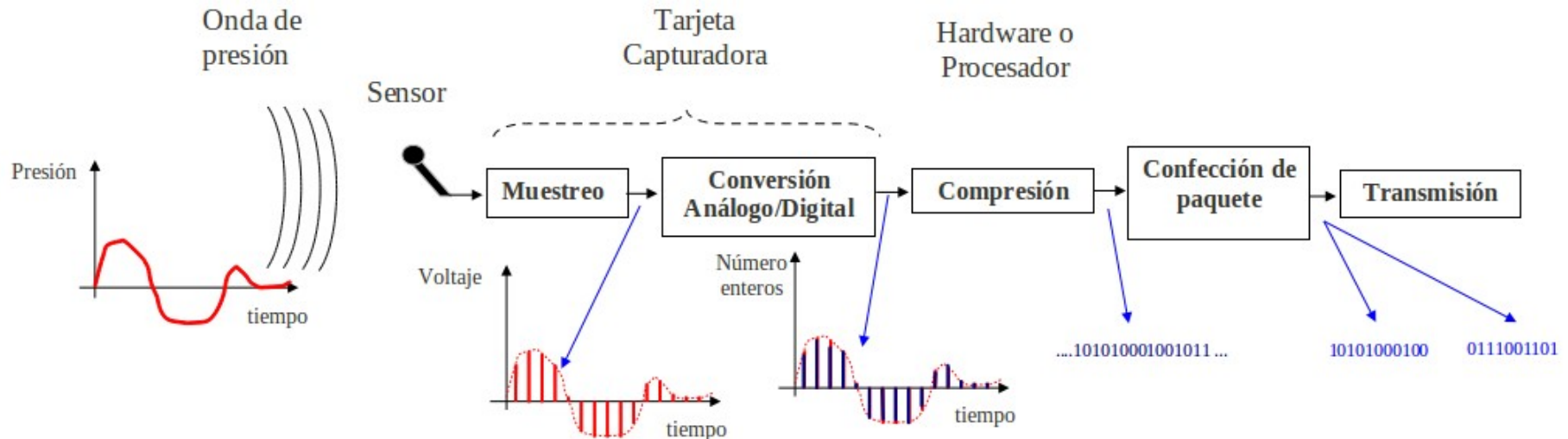
¿Cuál es tu opinión?

# Unas pocas palabras sobre audio en Redes de paquetes



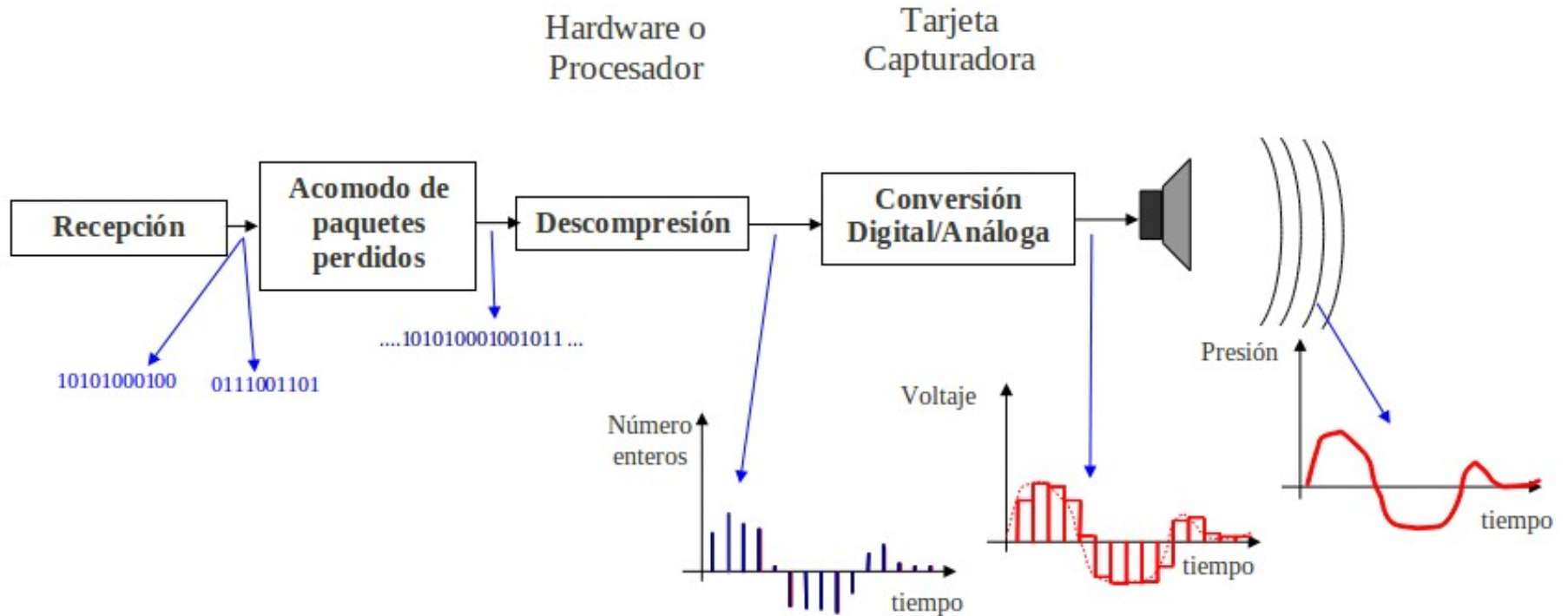
# Unas pocas palabras sobre audio en Redes de paquetes (Transmisión)

- Desde ondas de presión a unidades de transmisión (paquetes)
- Onda de presión → variación de voltaje → muestras discretas en tiempo → conversión análoga a digital (se discretiza en amplitud) → acumulación de muestras → compresión → creación de tramas → transmisión.



# Unas pocas palabras sobre audio en Redes de paquetes (Recepción)

- Desde unidades de transmisión a ondas de presión
- Recepción → acción ante pérdidas → descompresión → conversión digital a análoga → onda de presión



# Unas pocas palabras sobre compresión de audio

- ▣ Señal análoga muestreada a tasa fija
  - ▣ teléfono: 8,000 muestras/sec
  - ▣ CD musica: 44,100 muestras/sec
- ▣ Cada muestra es cuantizada, i.e., redondeada
  - ▣ e.g.,  $2^8=256$  valores posibles
- ▣ Cada valor cuantizado representado por bits
  - ▣ 8 bits => 256 valores
- ▣ Ejemplo: 8,000 muestras por segundo, 256 niveles --> 64,000 bps
- ▣ Receptor convierte a señal análoga:
  - ▣ Hay reducción de calidad

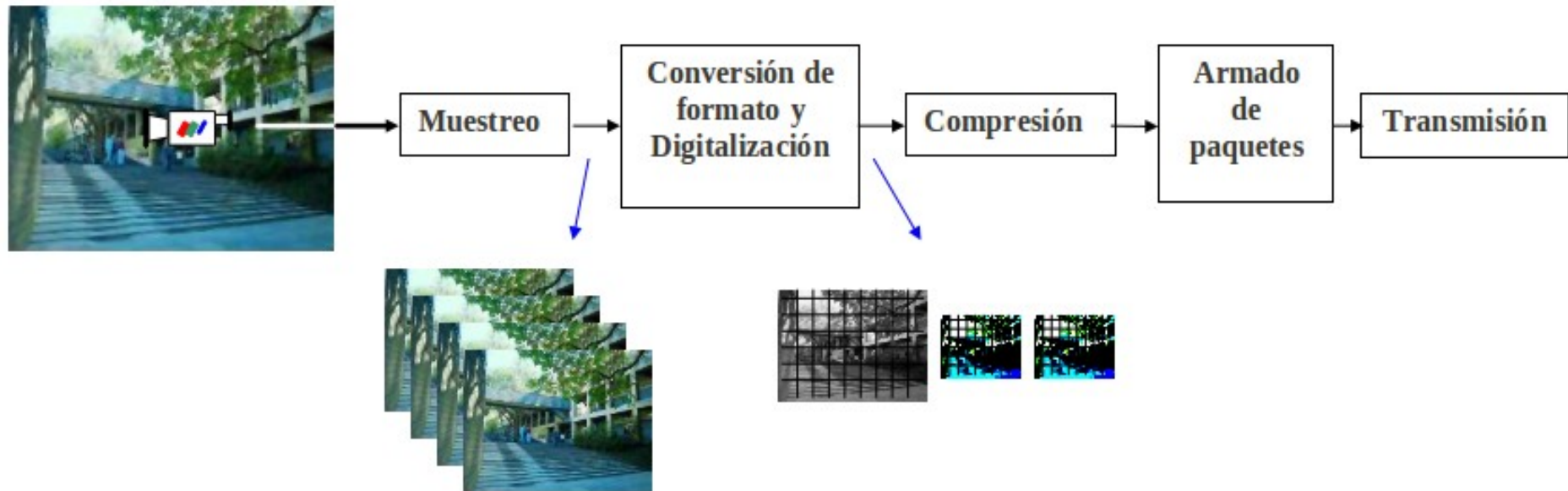
## Ejemplo tasas

- ▣ CD: 1.411 Mbps
- ▣ MP3: 96, 128, 160 kbps
- ▣ Telefonía en Internet: 5.3 - 13 kbps



# Unas pocas palabras sobre video en redes (Captura y transmisión)

- Luz → Secuencia de Imágenes (discretiza en espacio y tiempo) → Cambio de dominio de colores → Compresión → armado de paquetes → transmisión



- En la recepción se efectúa el proceso inverso
- Recepción -> reconocimiento de paquetes perdidos -> descompresión -> cambio de formato de colores -> despliegue

# Unas pocas palabras sobre compresión de vídeo

- ▣ Video es una secuencia de imágenes desplegadas a tasa constante
  - ▣ e.g. 24 imágenes/sec
- ▣ Imagen digital es un arreglo de pixeles
- ▣ Cada pixel es representado por bits
- ▣ Hay redundancia
  - ▣ Espacial (dentro de imagen)
  - ▣ Temporal (entre imágenes)

## Ejemplo:

- ▣ MPEG 1 (CD-ROM) 1.5 Mbps
- ▣ MPEG2 (DVD) 3-6 Mbps
- ▣ MPEG4 (común en Internet, < 1 Mbps)

## Investigación:

- ▣ Video en capas (escalable)
  - ▣ adapta capas a BW disponible

# Capítulo 7: Contenidos

- ▣ 7.1 Aplicaciones Multimedia en Red
- ▣ 7.2 Streaming de Audio y video almacenado
- ▣ 7.3 Real-time Multimedia: Estudio de telefonía en Internet
- ▣ 7.4 protocolos para aplicaciones Interactivas de Tiempo Real
  - ▣ RTP, RTCP, SIP
- ▣ 7.5 Distribución de Multimedia: Redes de distribución de Contenidos
- ▣ 7.6 Más allá de Best Effort
- ▣ 7.7 Mecanismos de itineración y políticas
- ▣ 7.8 Servicios Integrados y Servicios Diferenciados
- ▣ 7.9 RSVP

# Streaming de Multimedia Almacenada

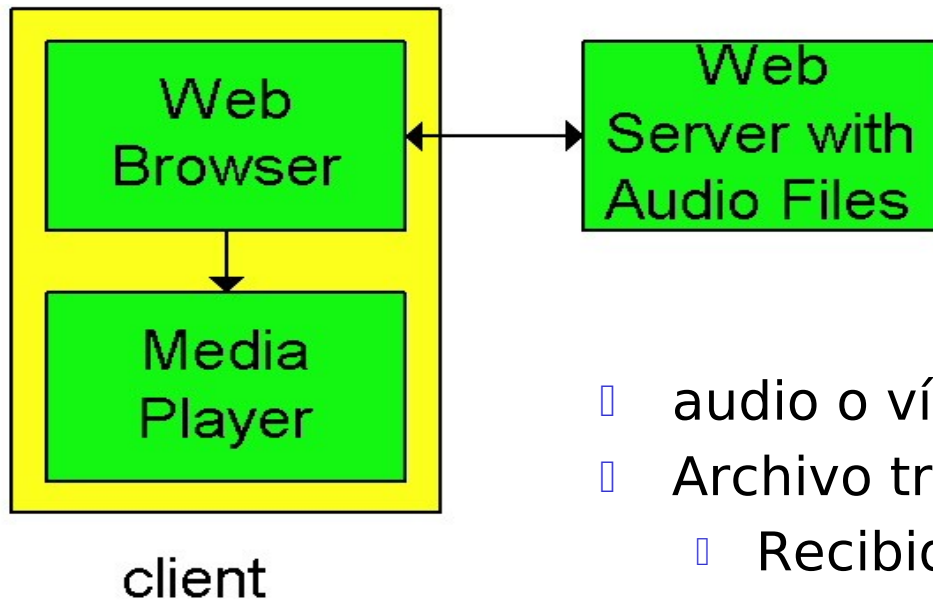
Streaming a nivel de aplicación para obtener lo mejor del servicio “best effort” :

- Buffering en lado cliente
- uso de UDP versus TCP
- codificación múltiple de multimedia

## Reproductor

- Remover jitter
- Descompresión
- Acomodo por errores
- Interfaces gráficas con control de interactividad

# Multimedia en Internet: caso más simple

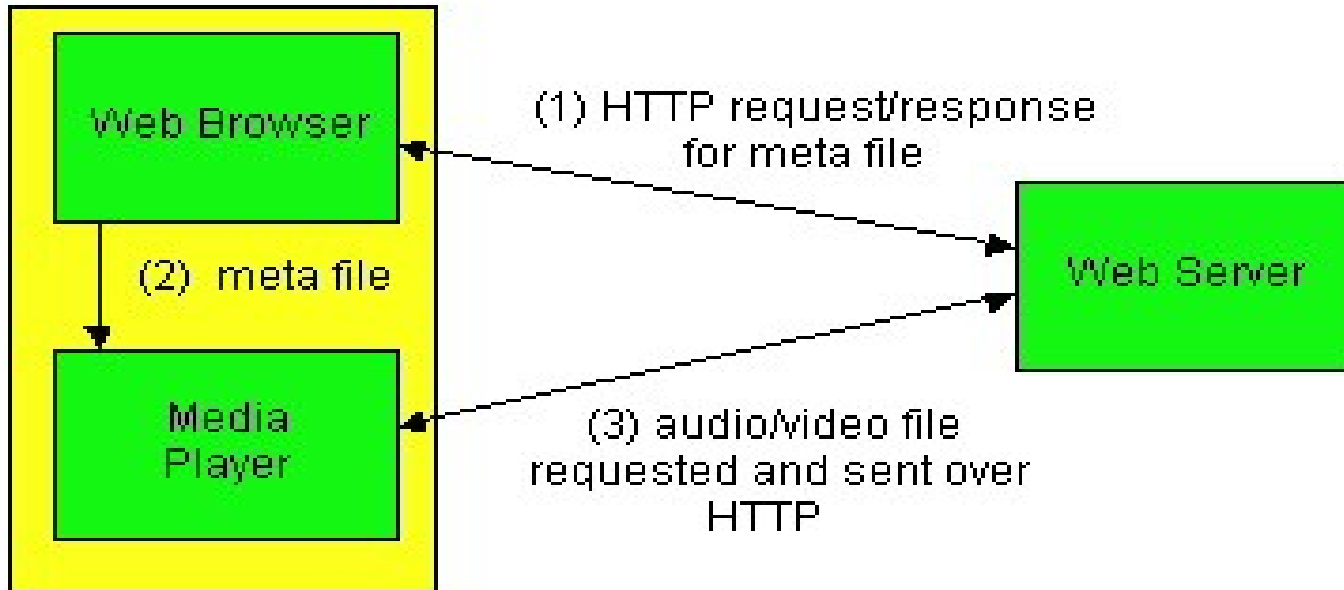


- audio o vídeo almacenado en archivo
- Archivo transferido como objeto HTTP
  - Recibido enteramente en cliente
  - Entonces es pasado al reproductor

**audio, vídeo no es flujo continuo:**

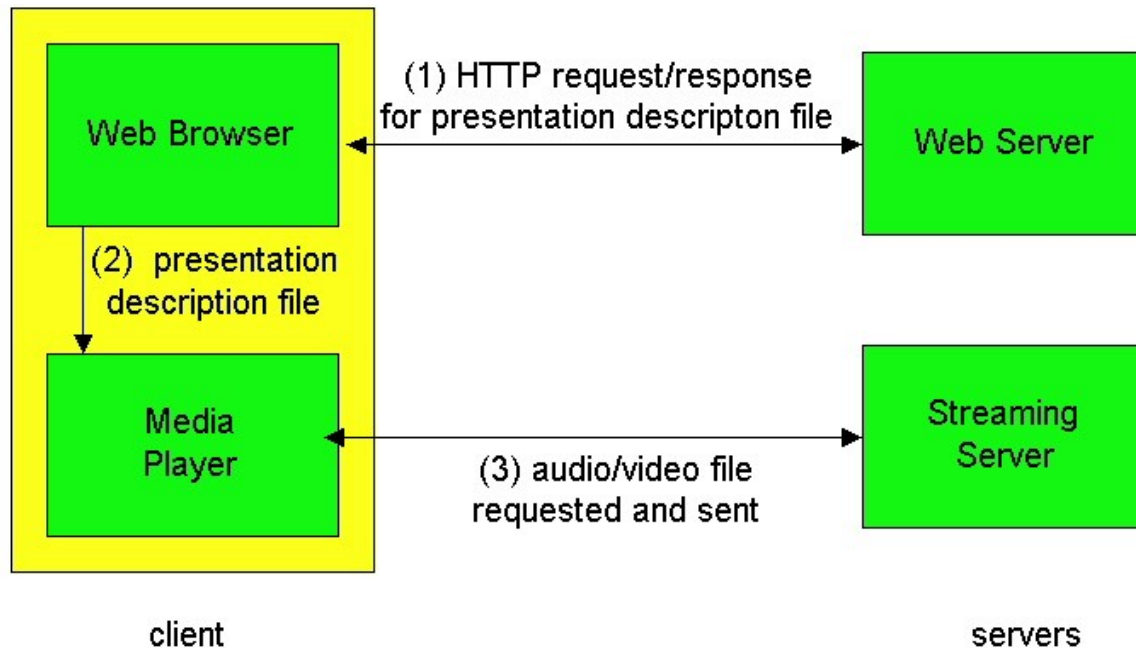
- no, “pipelining,” gran retardo hasta reproducción!

# Multimedia en Internet: Vía streaming



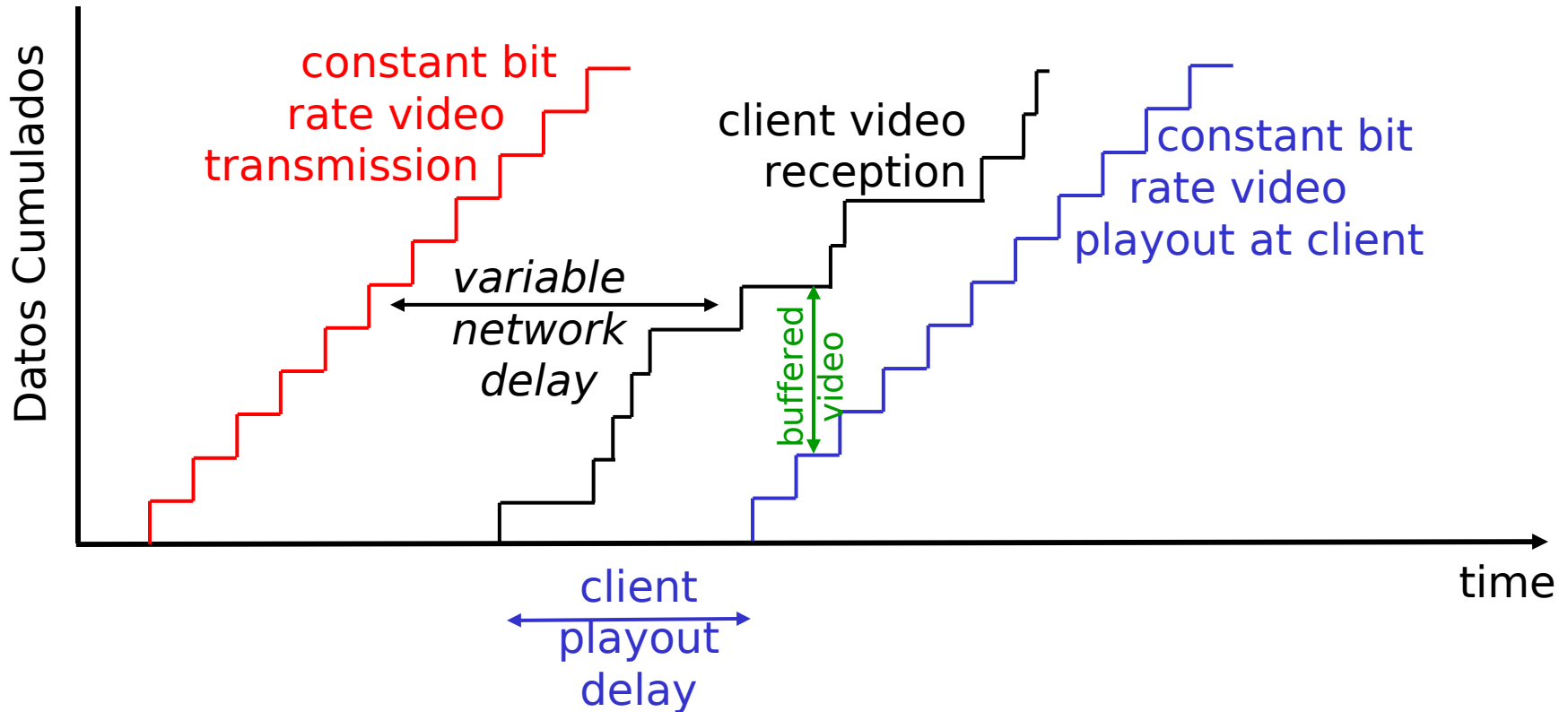
- ▮ Navegador Obtiene **metafile**
- ▮ Navegador lanza el reproductor pasando el metafile
- ▮ Reproductor contacta al servidor
- ▮ Servidor envía flujo (**stream**) de audio/vídeo a reproductor

# Streaming desde servidor de streaming



- Esta arquitectura permite protocolo distintos a HTTP entre servidor y reproductor
- Puede usar UDP en lugar de TCP en paso (3).

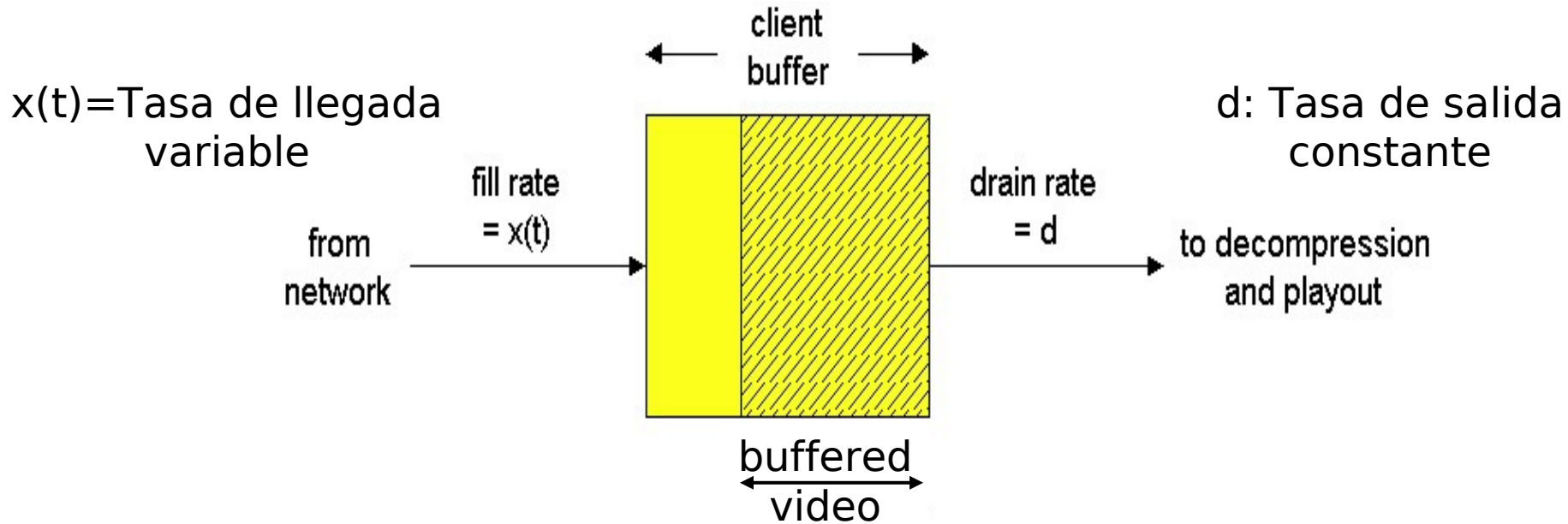
# Streaming de Multimedia: Buffering en Cliente



- Buffering en lado cliente, retardo en reproducción compensa variaciones de retardo de la red



# Streaming de Multimedia: Buffering en cliente



- Buffering en lado cliente, retardo en reproducción compensa variaciones de retardo

# Streaming de Multimedia: UDP o TCP?

## UDP

- ▣ Servidor envía a tasa apropiada para cliente (ignora congestión de red !)
  - ▣ Tasa envío = tasa de codificación = tasa constante
  - ▣ entonces, tasa llegada = tasa cte. - tasa pérdida
- ▣ Retardo de reproducción pequeño (2-5 segundos) para compensar variaciones de retardo en red
- ▣ Recuperación de errores: lo que el retardo permita

## TCP

- ▣ Enviar a tasa máxima posible bajo TCP
- ▣ Llegada de paquetes fluctúa debido a control de congestión de TCP
- ▣ Retardo de reproducción mayor: suaviza tasa de arribo variable de TCP
- ▣ HTTP/TCP pasa más fácilmente a través de firewalls

# Streaming de Multimedia: Nuevos protocolos: DCCP y SCTP

## DCCP: Datagram CongestionControl Protocol

- ▣ Hecho estándar en Marzo de 2006. Su primera implementación versión 2.6.14 de Linux (Octubre 2005)
- ▣ Como UDP, es orientado a mensajes (o bloques).
- ▣ Implementa establecimiento y término de conexión confiable.
- ▣ Implementa Notificaciones explícitas de congestión y control de congestión.

## SCTP: Stream Control Transmission Protocol

- ▣ Hecho estándar en Septiembre de 2007. También existen implementaciones de él en Linux.
- ▣ Es orientado a mensajes como UDP, asegura confiabilidad, orden en el mensaje y control de congestión como TCP.

# Streaming de Multimedia: tasa(s) en cliente



**Q:** Cómo manejar capacidades diferentes de recepción en clientes?

- ▮ 384 Kbps comunicación 3G inalámbrica
- ▮ 100Mbps Ethernet

**A:** servidor almacena, transmite múltiples copias de vídeo, codificadas a distinta tasa

# Control de usuario del Streaming del Medio: Real-time Streaming Protocol (RTSP)

## HTTP

- No apunta a contenido multimedia
- No hay comandos para fast forward, etc.

## RTSP: RFC 2326

- Protocolo capa aplicación entre Cliente y servidor.
- Para que usuario controle despliegue: rewind, fast forward, pause, resume, saltos, etc...

## Qué no hace:

- No define cómo audio/vídeo es encapsulado para su envío paulatino (streaming) en la red
- No indica cómo el flujo es transportado en la red; puede ser sobre UDP o TCP
- No especifica cómo el reproductor “bufferea” audio/vídeo

# RTSP (Real-time streaming protocol): control fuera de banda

## FTP usa un canal de control “out-of-band”:

- Un archivo es transportado sobre una conexión TCP.
- Información de control (cambios de directorio, borrar/renombrar archivos, etc.) es enviada sobre conexión TCP separada.
- Los canales “out-of-band” e “in-band” usan número de puertos diferentes.

## Mensajes RTSP también son enviados out-of-band:

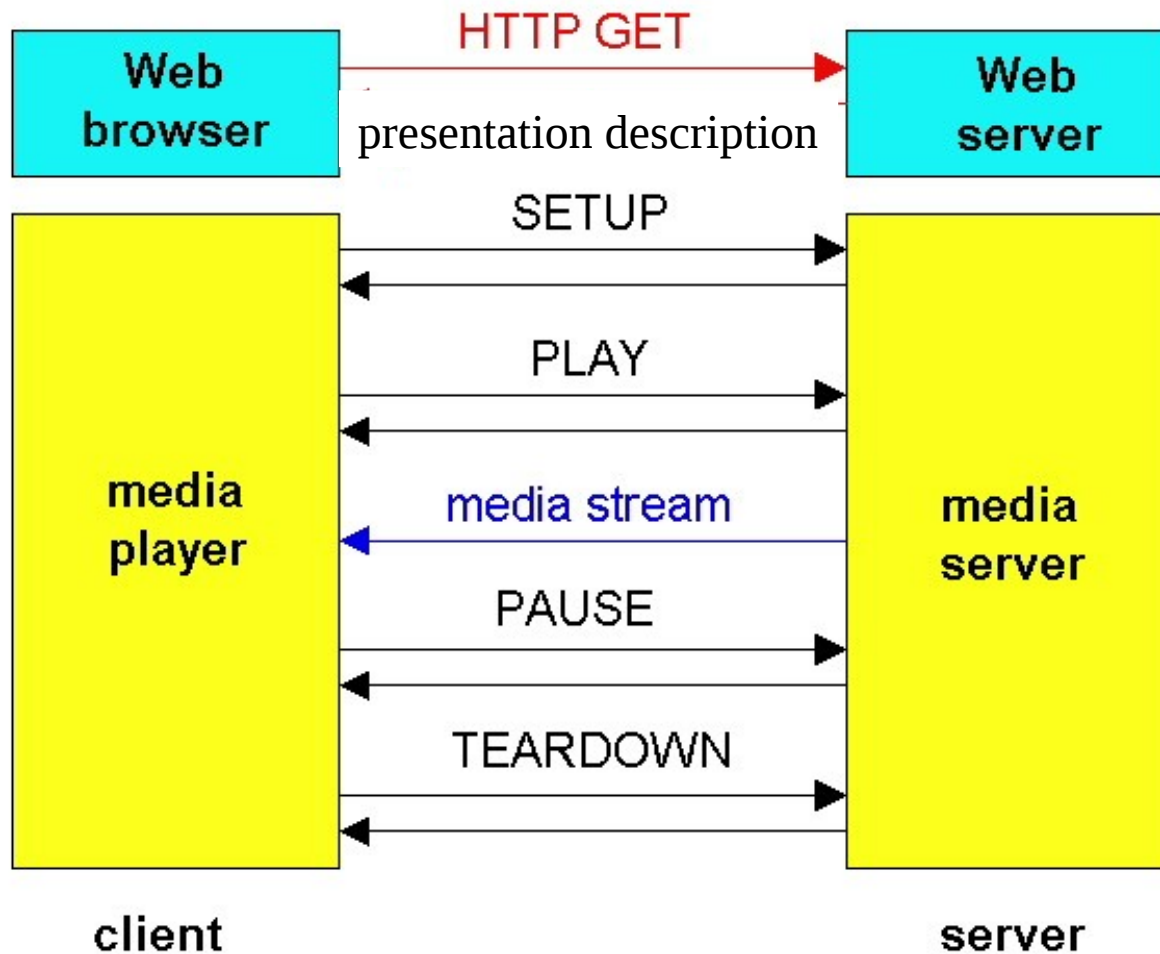
- Mensajes de control de RTSP usan número de puerto distinto a flujo de media: out-of-band.
  - Port 554
- El flujo del medio (audio, video u otro) es considerado “in-band”.

# Ejemplo RTSP

## Escenario:

- metafile es comunicado al navegador web
- Navegador lanza reproductor
- Reproductor configura una conexión de control RTSP y conexión de datos al servidor de streaming

# Operación de RTSP





# Ejemplo de Metafile

```
<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>
```

# Ejemplo de intercambio 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
```

# Capítulo 7: Contenidos

- ▣ 7.1 Aplicaciones Multimedia en Red
- ▣ 7.2 Streaming de Audio y video almacenado
- ▣ 7.3 Real-time Multimedia: Estudio de telefonía en Internet
- ▣ 7.4 protocolos para aplicaciones Interactivas de Tiempo Real
  - ▣ RTP, RTCP, SIP
- ▣ 7.5 Distribución de Multimedia: Redes de distribución de Contenidos
- ▣ 7.6 Más allá de Best Effort
- ▣ 7.7 Mecanismos de itineración y políticas
- ▣ 7.8 Servicios Integrados y Servicios Diferenciados
- ▣ 7.9 RSVP