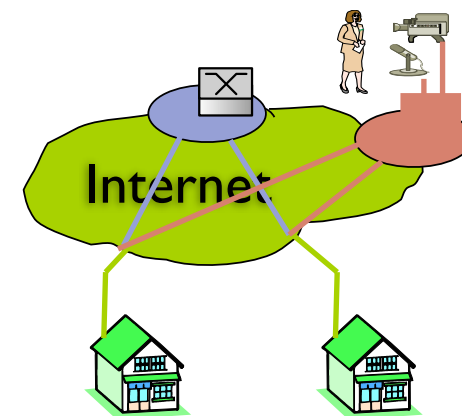
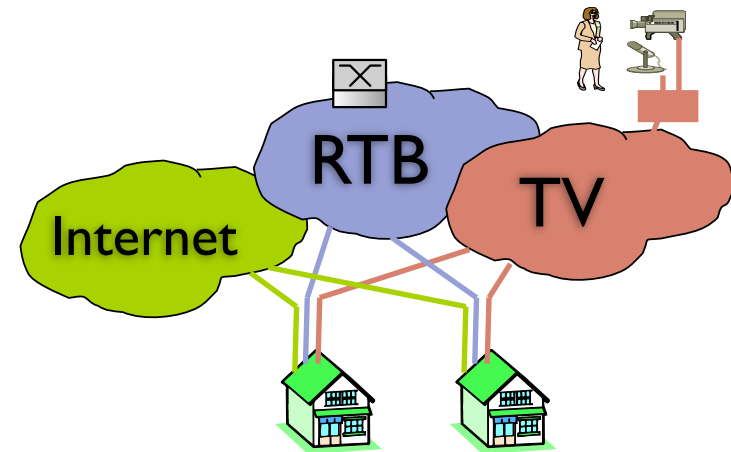


Transporte de Vídeo

Introducción

- ▶ Hasta hace poco...
 - Red de datos basada en TCP/IP
 - Red telefónica independiente
 - Distribución de TV independiente

- ▶ Todo sobre la red de datos
 - De momento va convergiendo pero sobre todo la red de acceso
 - TV por cable + internet + telefono
 - ADSL + TV + telefono
 - Pero hasta que punto estan transportando la TV y el teléfono por sus propias redes
 - Al usuario le llegue un solo cable a su casa
“Triple play”
- ▶ En clases anteriores como integrar el audio
- ▶ Hoy: video



Multimedia Networking Applications

Características principales:

- ▶ Sensibles al retardo
 - end-to-end delay
 - delay jitter
- ▶ Toleran pérdidas: si son infrecuentes solo causan pequeños *glitches*
- ▶ Requisitos diferentes de la transferencia de datos (ficheros)

Clase de aplicaciones multimedia

- “Streaming” de audio o vídeo grabado
- Streaming de audio o vídeo en vivo
- Audio o vídeo interactivo

Jitter es la variabilidad de los retardos dentro del mismo flujo de paquetes

Servicios multimedia sobre la red de datos

AUDIO

- ▶ Telefonía sobre Internet
IP-telephony
 - best-effort no es suficiente
 - QoS y planear la red apropiadamente

VIDEO (+AUDIO)

- ▶ Televisión sobre Internet
IPTV
 - QoS
 - Mucho más BW
 - Pero...
Punto a multipunto
- ▶ Más allá de la TV
Video on demand
 - Mismas dificultades
 - Sin las ventajas del envío a multipunto

Vídeo digital

- ▶ Dibujar una imagen detrás de otra da sensación de movimiento...

- ¿Cómo de rápido? Del orden de al menos 25 imágenes por segundo (aunque luego el monitor refresque más rápido para evitar flicker)

- ▶ Cada fotograma (frame) es una imagen digital a transmitir

Su calidad depende de

- Resolución (pixels)
 - Colores (cuantos bits para representar el color, RGB o luminancia/crominancia)

- ▶ Cuantos frames por segundo vamos a transmitir?

- ▶ Ejemplo:

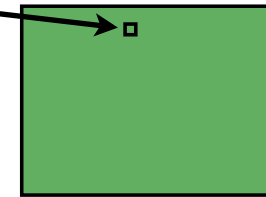
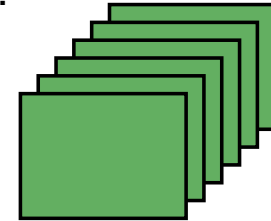
- Video analógico en PAL

625 líneas (576 visibles) aspecto 4:3 25 fps

Usamos 768x576 pixels, en cada pixel representamos 16 millones de colores con 8:8:8 bits para RGB

Serían 1327kB cada frame o sea **265Mbps** para enviar **un solo canal de TV**

- ▶ Es muy importante usar técnicas de compresión en los formatos de vídeo digital



Formatos

Moving Picture Experts Group de ISO/IEC

Formatos de video comprimido:

▶ MPEG-1

- Entornos libres de errores (CD-ROM, VCD)
- 1.2 Mbps el vídeo, 256 Kbps el audio
- Busca la mejor calidad dado un bitrate (tasa de lectura CD-ROM 1.416Mbps)
- Permite acceso aleatorio a un frame
- MPEG-1 part 3 define el audio 3 tipos, el tercero es MPEG-1 part3 layer3 : MP3
- Basado en los trabajos para el H.261 de ITU, imagen dividida en macrobloques y DCT de los bloques

▶ MPEG-2

- Broadcast TV (DVB) y DVD
- 2-15 Mbps (vídeo+audio)
- Mantiene calidad constante
- MPEG-2 part 7 advanced audio codec AAC
- Basado en DCT pero con técnicas de compresion basadas en correlacion temporal... diferente compresión segun el movimiento de la imagen
- Define encapsulado (transport stream TS) para enviar el video sobre la red

ITU-T H.26x

- ▶ **ITU-T H.261** “Video codec for audiovisual services at px64 kbits”
 - CIF (352x288), QCIF (176x144) (resoluciones de luminancia)
 - El resultado es un bit stream
 - Videoconferencia sobre ISDN

- ▶ **ITU-T H.263** “Video coding for low bit rate communication”
 - Basado en H.261
 - sub-QCIF (128x96) , QCIF, CIF, 4CIF (704x576) and 16CIF (1408x1152)
 - Videoconferencia sobre POTS
 - Basado en H.261 MPEG-1 y MPEG-2

ITU-T H.262

- ▶ “Information technology - Generic coding moving pictures and associated audio information: Video”
- ▶ Vídeo como MPEG-2 Part 2
- ▶ Pensando en vídeo sobre ATM y HDTV
- ▶ Soporta *scalable video encoding* (ej: codificación espacial escalable mediante varias capas que van refinando la imagen)
- ▶ Tipos de frames:
 - Intra Coded Pictures (I-Pictures): sin referencia a otras imágenes
 - Predictive Coded Pictures (P-Pictures): emplea *motion compensated prediction* con la anterior I- o P- Picture
 - Bidirectional-predictive Coded Pictures (B-Pictures): relativas a anterior y posterior I- o P- Picture
 - La organización en secuencia es flexible en el estándar

H.264

- ▶ “Advanced video coding for generic audiovisual services”
- ▶ MPEG-4 Part 10
- ▶ Puede dar la misma calidad que codec de vídeo de MPEG-2 con la mitad o un tercio del bitrate

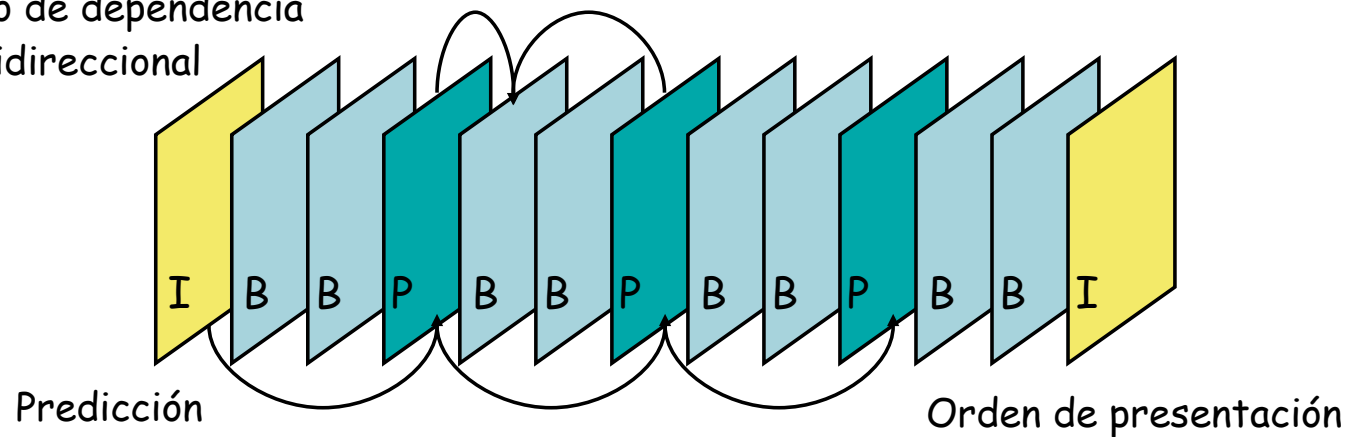
Use Scenario	Resolution & Frame Rate	Example Data Rates
Mobile Content	176x144, 10-15 fps	50-60 Kbps
Internet/Standard Definition	640x480, 24 fps	1-2 Mbps
High Definition	1280x720, 24p	5-6 Mbps
Full High Definition	1920x1080, 24p	7-8 Mbps

GoP

- *Group of Pictures*
 - ▶ Típicamente 1/2 sec cada GoP
 - ▶ Orden
 - De presentación
Ej.: IBBPBBPBBPBB ibbpbb...
 - De codificación
Ej.: I bb PBBPBBPBB i BB pbb...
 - ▶ Closed or Open GoP
 - ▶ Broken GoP: falta el GoP anterior

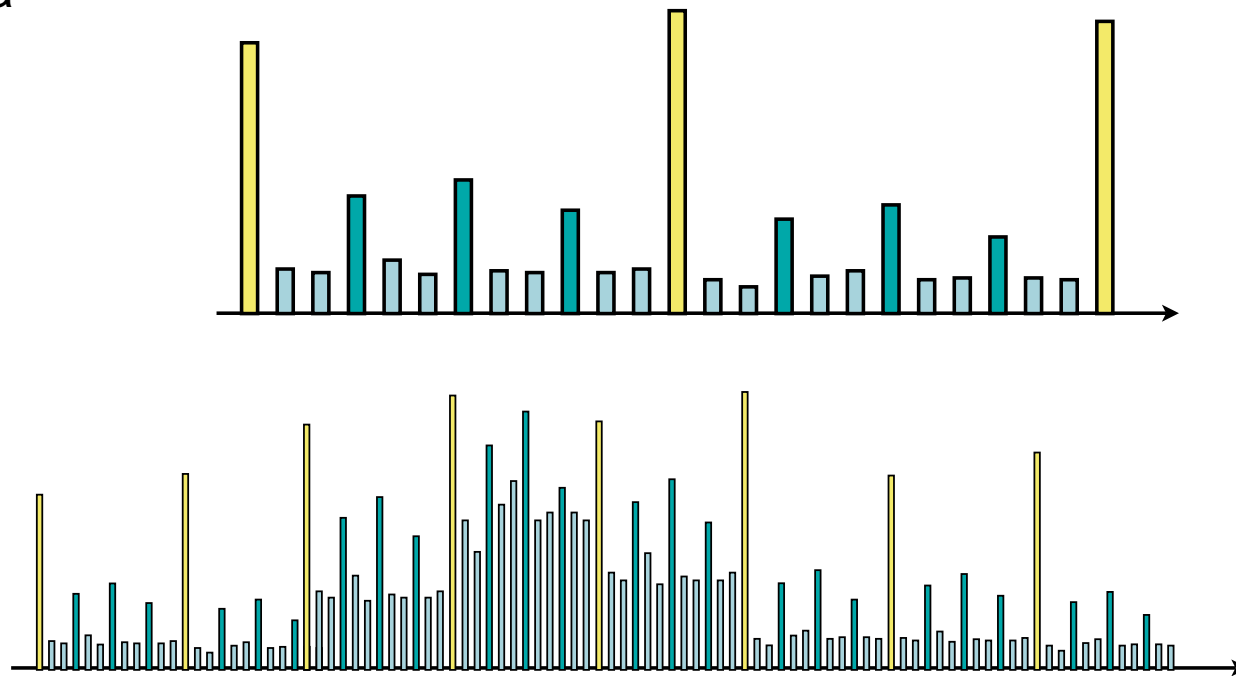
	I frame	P frame	B frame
Compression Ratio	Low	Good	Best
Random Access	Best	Hard	Hardest
Complexity	Normal	High	Highest

Ejemplo de dependencia
bidireccional



GoP

- ▶ La cantidad de información a transmitir por cada frame es variable
- ▶ Las tramas I contienen más información
- ▶ Las tramas P y B más pequeñas (mayor compresión)
- ▶ Variabilidad de tamaño en escalas pequeñas por las tramas
- ▶ Variabilidad de tamaño en escalas mayores según el movimiento de la escena



Tasa de bits para la red

▶ Constant Bit Rate

- Calidad y tasa de compresion varian para conseguir una tasa de datos constante

▶ Ejemplo: MPEG-1

▶ Ventajas

- Recursos previsibles

▶ Desventajas

- Se sacrifica la calidad cuando hay cambios rápidos en la imagen (escenas con movimiento)
- Se podría haber comprimido más cuando hay poco movimiento

▶ Variable Bit Rate

- Bit rate y tasa de compresion pueden variar para conseguir un nivel de calidad constante

▶ Ejemplo: MPEG-2

▶ Ventajas

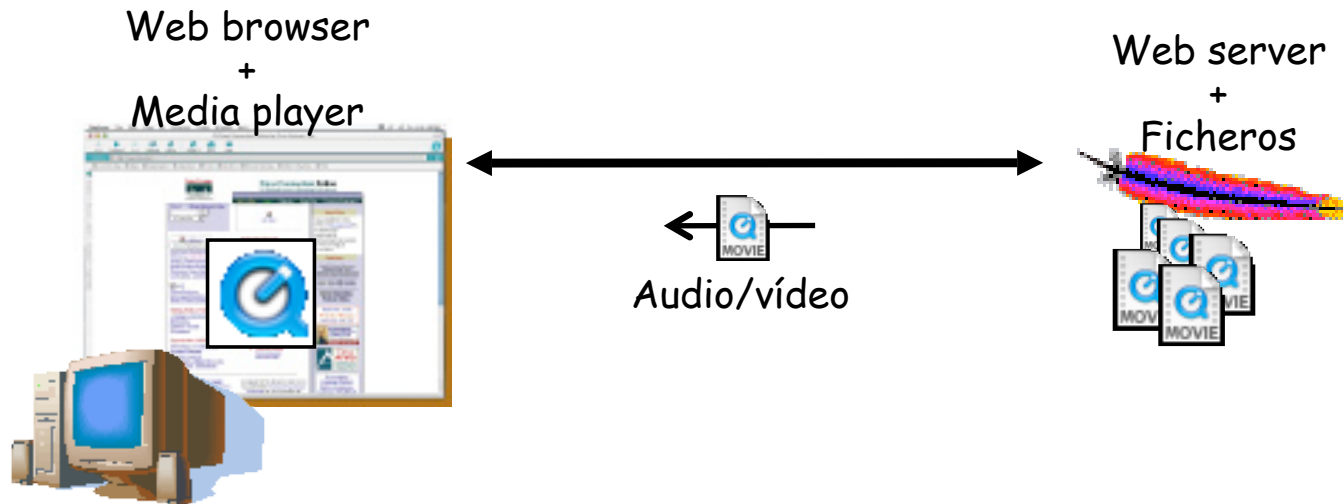
- Se mantiene la calidad objetivo sin desperdiciar recursos
- Multiplexación estadística

▶ Desventajas

- Necesidades de recursos más difíciles de prever
- Sistemas más complicados

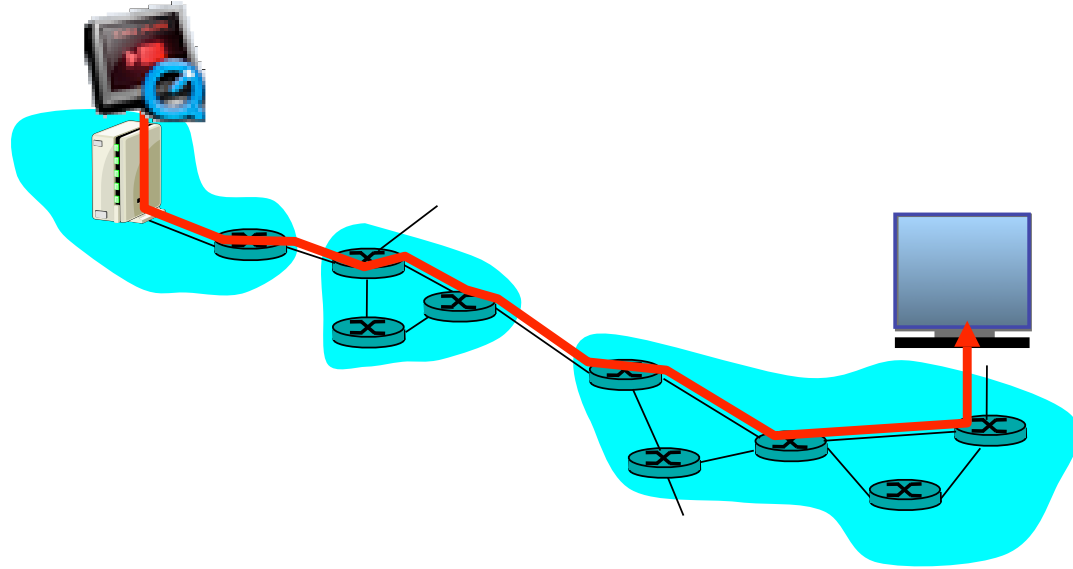
Servicios de video

Download+play vs “Streaming”



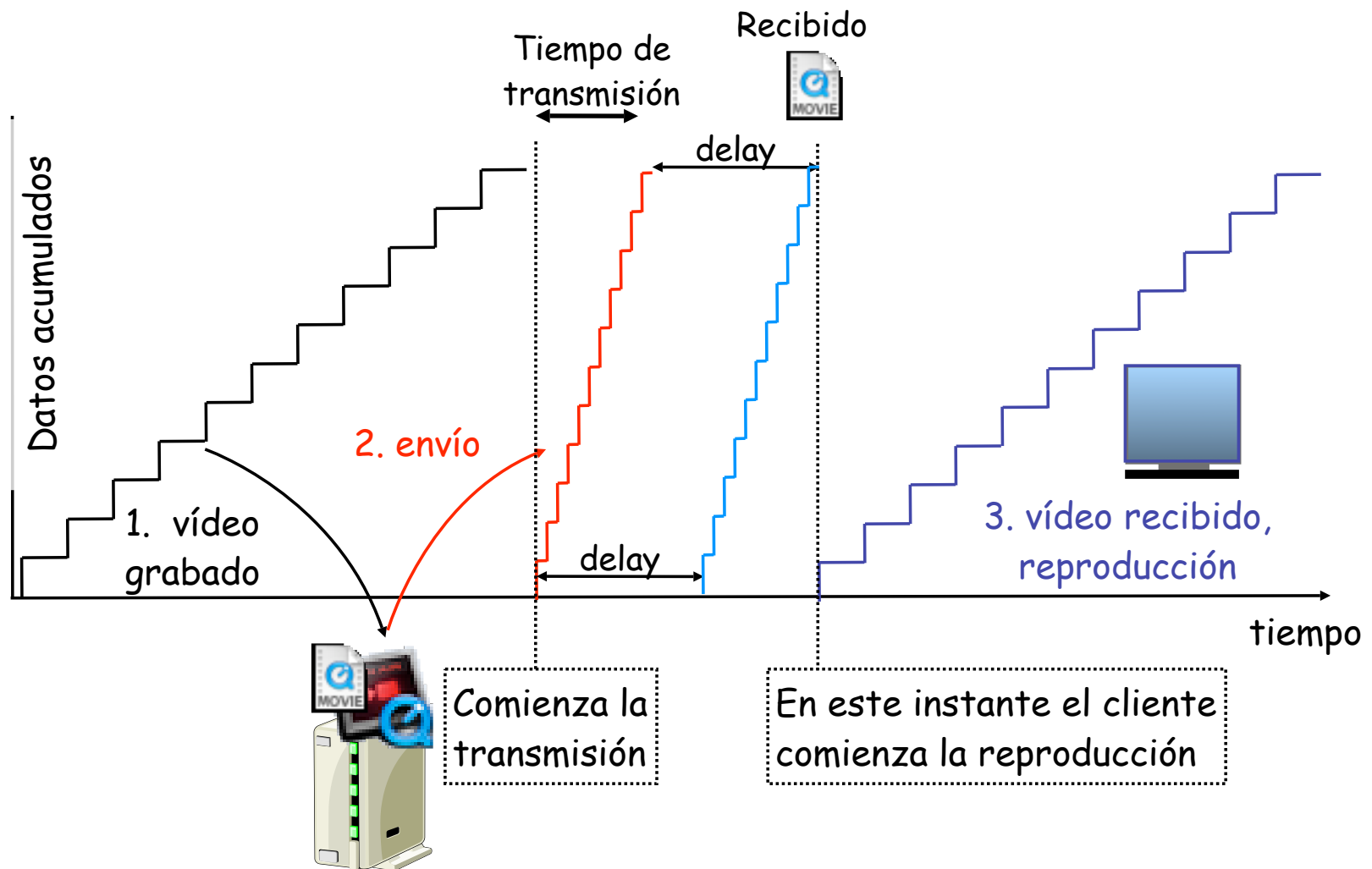
- Ficheros de audio o vídeo almacenados en servidor
- Ficheros transferidos como objetos HTTP
 1. Download+play
 - Recibidos completamente por el cliente
 - Pasados al reproductor
 - No hay “streaming”. Gran espera hasta empezar
 2. “Streaming”
 - Pasados al reproductor a medida que los recibe
 - Calcula cuándo empezar dada la velocidad a la que recibe
 - “Streaming” pero recibe a la máxima velocidad (transferencia HTTP)

“Streaming” de multimedia en disco

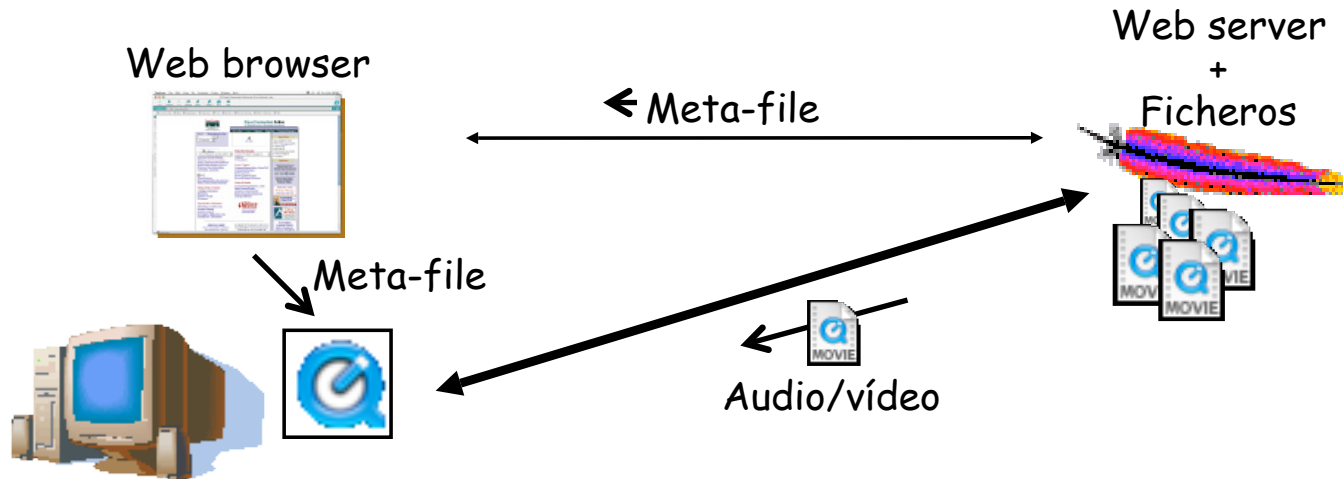


- ▶ “Streaming”
 - El cliente comienza la reproducción antes de recibir todos los datos
 - Es más bien un “play-as-you-get”
 - Requisitos temporales para los datos aún por recibir
- ▶ Streaming
 - La velocidad de transmisión es aproximadamente la de consumir los datos

Download+play

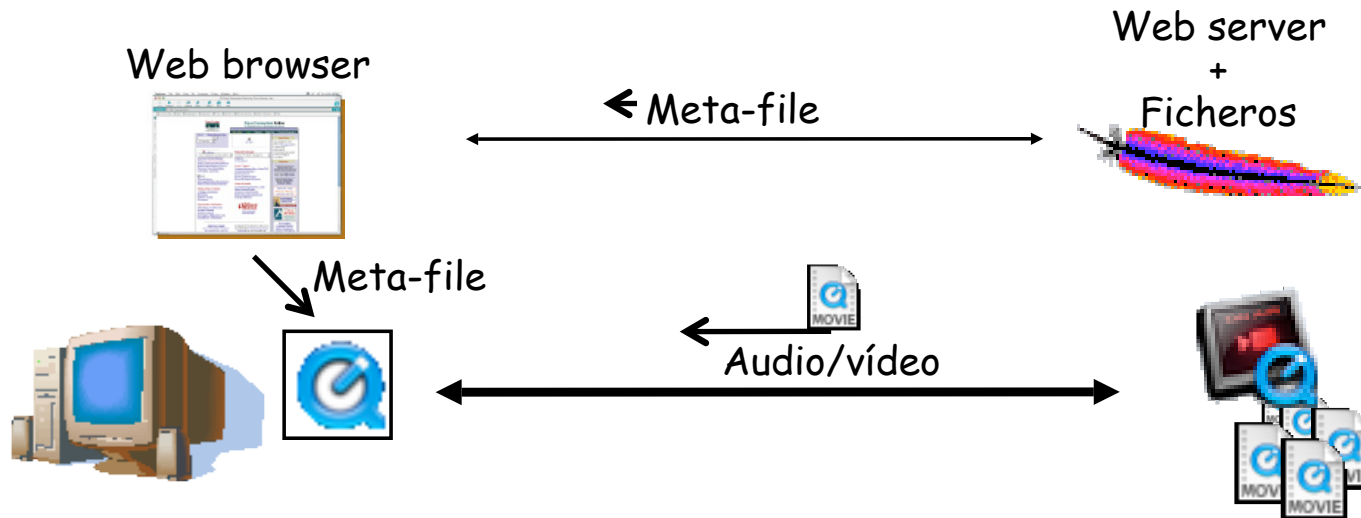


Play-as-you-get



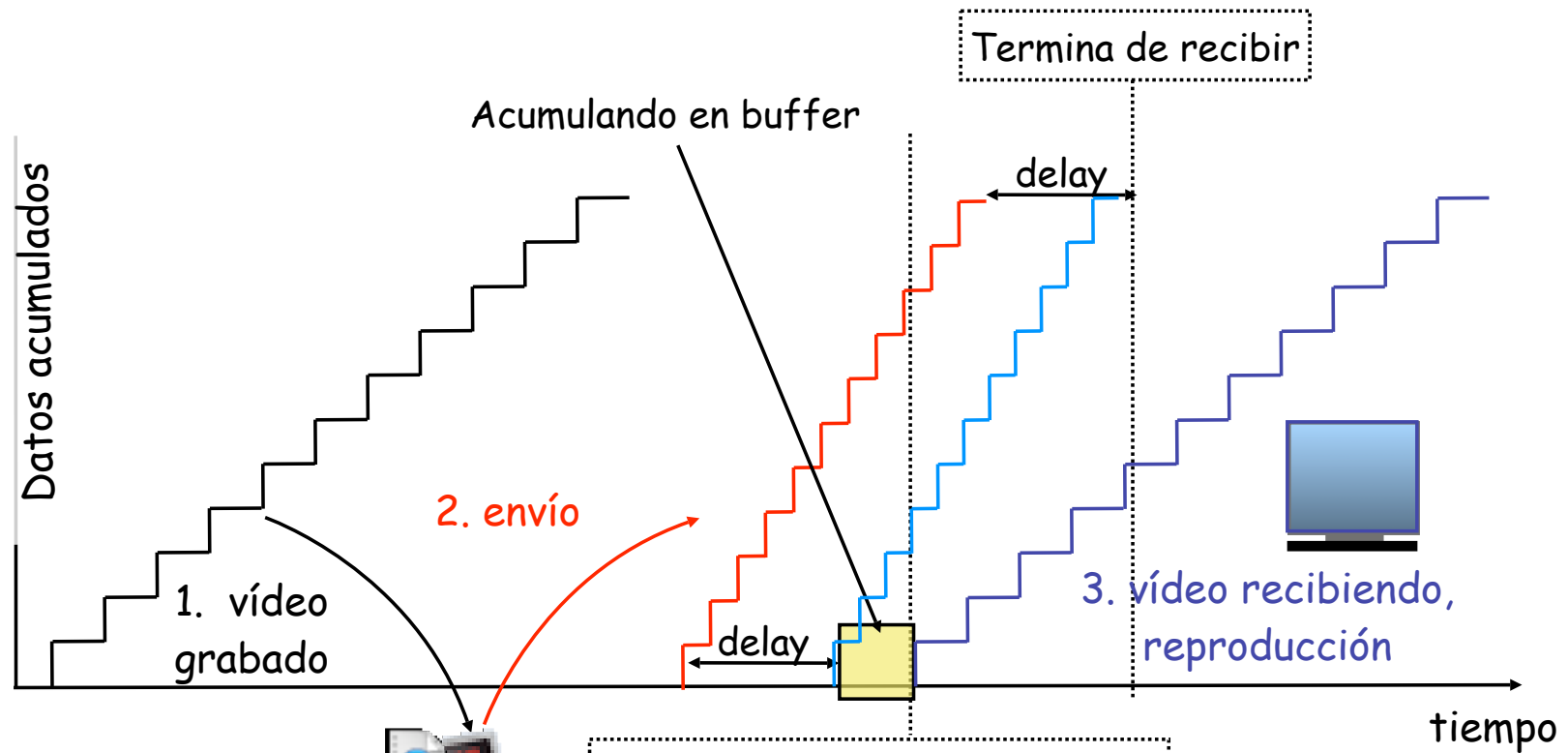
- Ficheros de audio o vídeo
- El navegador obtiene un *meta-file* que describe cómo conseguir la película
- Le pasa ese *meta-file* al reproductor
- El reproductor contacta con el servidor y solicita el fichero
- El servidor se lo envía (HTTP)

Play-as-you-get



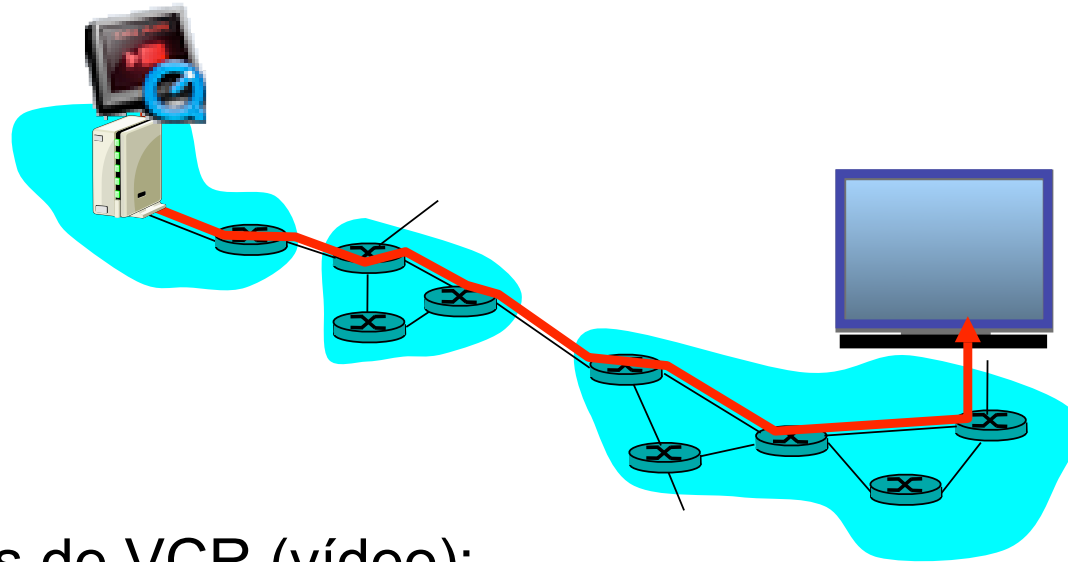
- ▶ Permite que se emplee un protocolo diferente de HTTP
- ▶ Permite emplear UDP en vez de TCP

Play-as-you-get



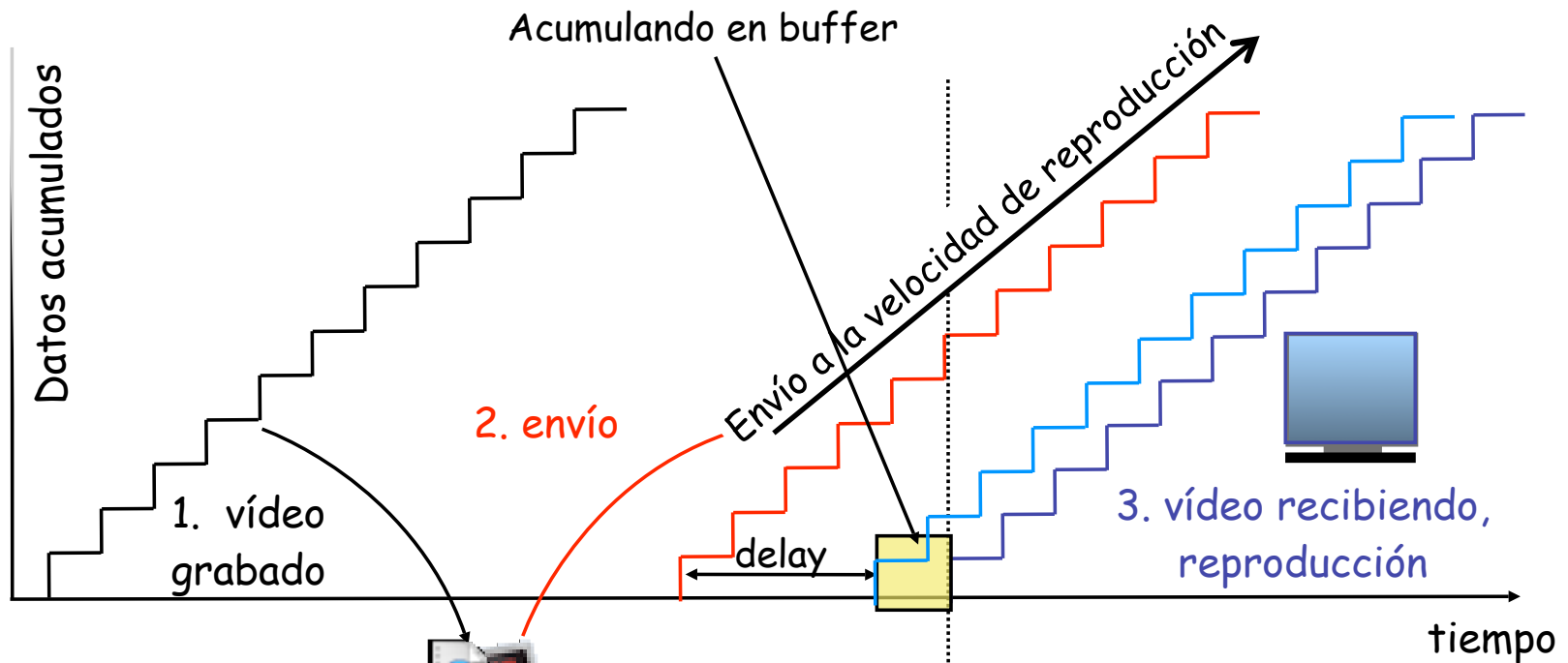
En este instante el cliente comienza la reproducción mientras el servidor continúa transmitiendo lo que falta

“Streaming” de multimedia en disco: Interactividad



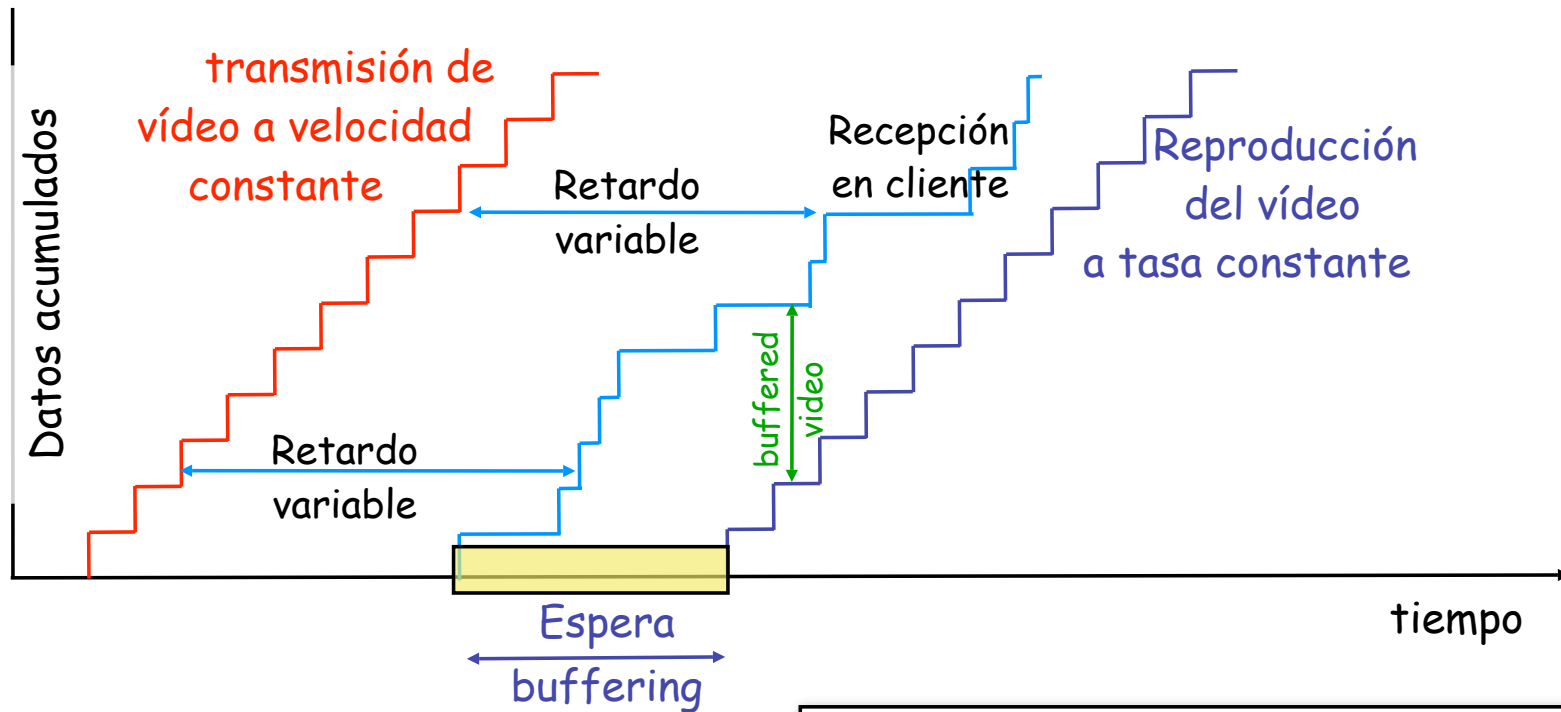
- ▶ Funcionalidades de VCR (vídeo):
 - Pausa, rebobinar, avance rápido, etc.
 - Retardo de comienzo 10 segs OK
 - Retardo ante un comando 1-2 seg OK
 - Protocolos para estos comandos (RTSP)

Streaming

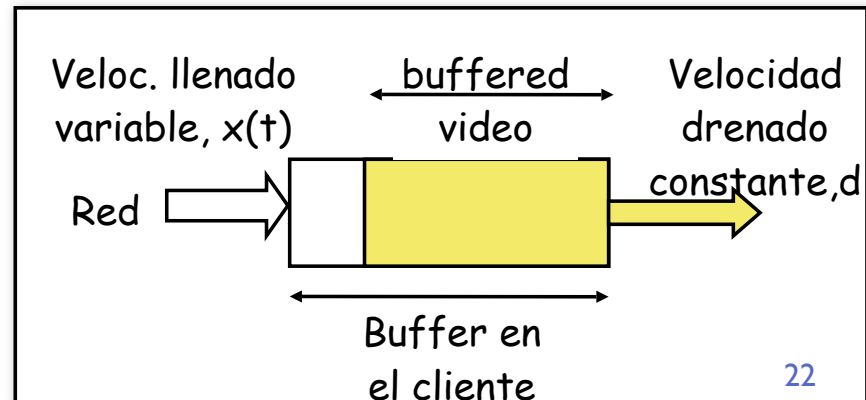


En este instante el cliente comienza la reproducción mientras el servidor continúa transmitiendo lo que falta

Streaming: Client Buffering

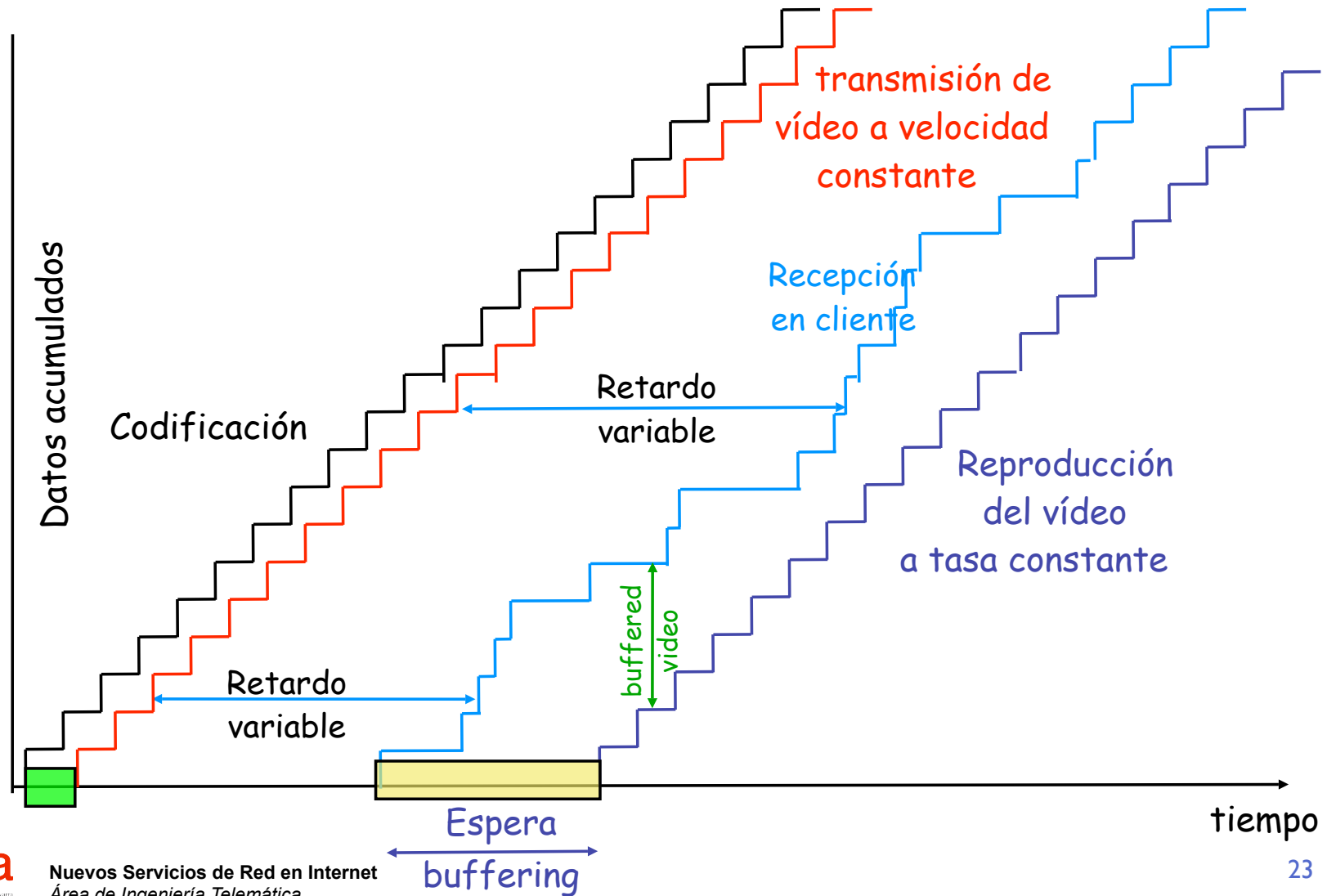


- ▶ El cliente acumula parte del vídeo en buffer antes de comenzar
- ▶ Permite absorber variaciones en el retardo en la red (*jitter*)
- ▶ Si el buffer se vacía el reproductor se detiene



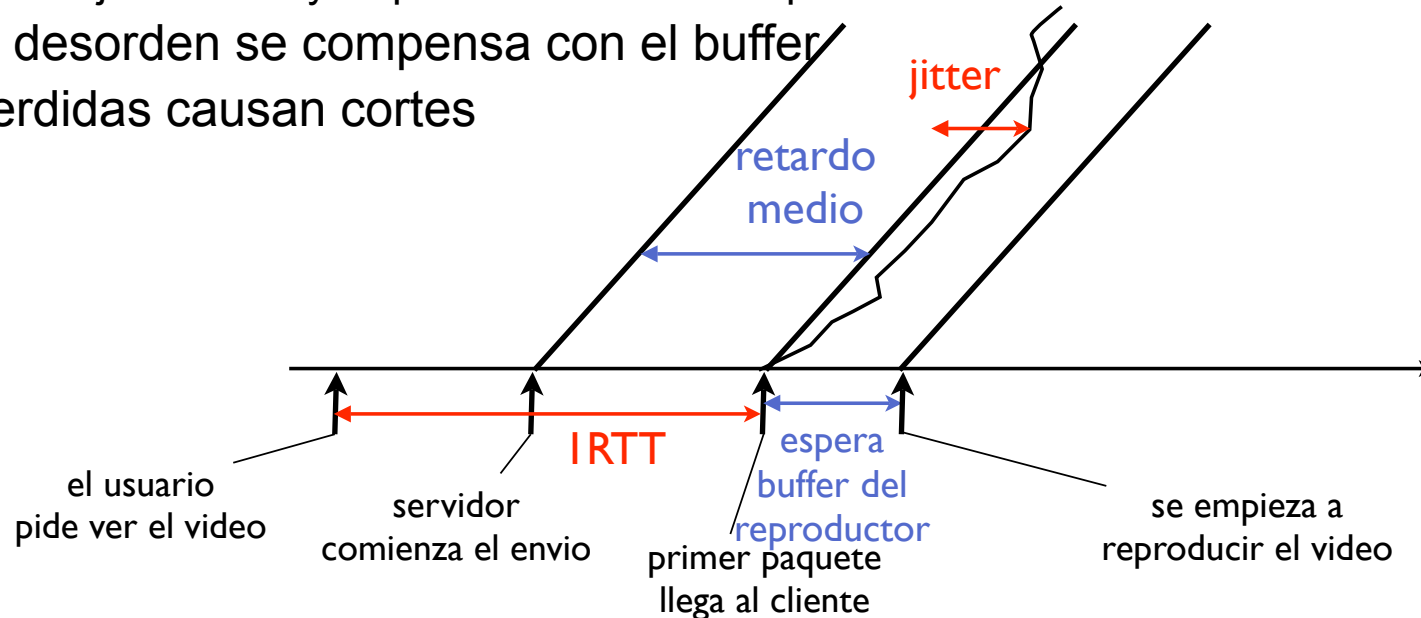
Live Streaming

- ▶ Avance rápido imposible



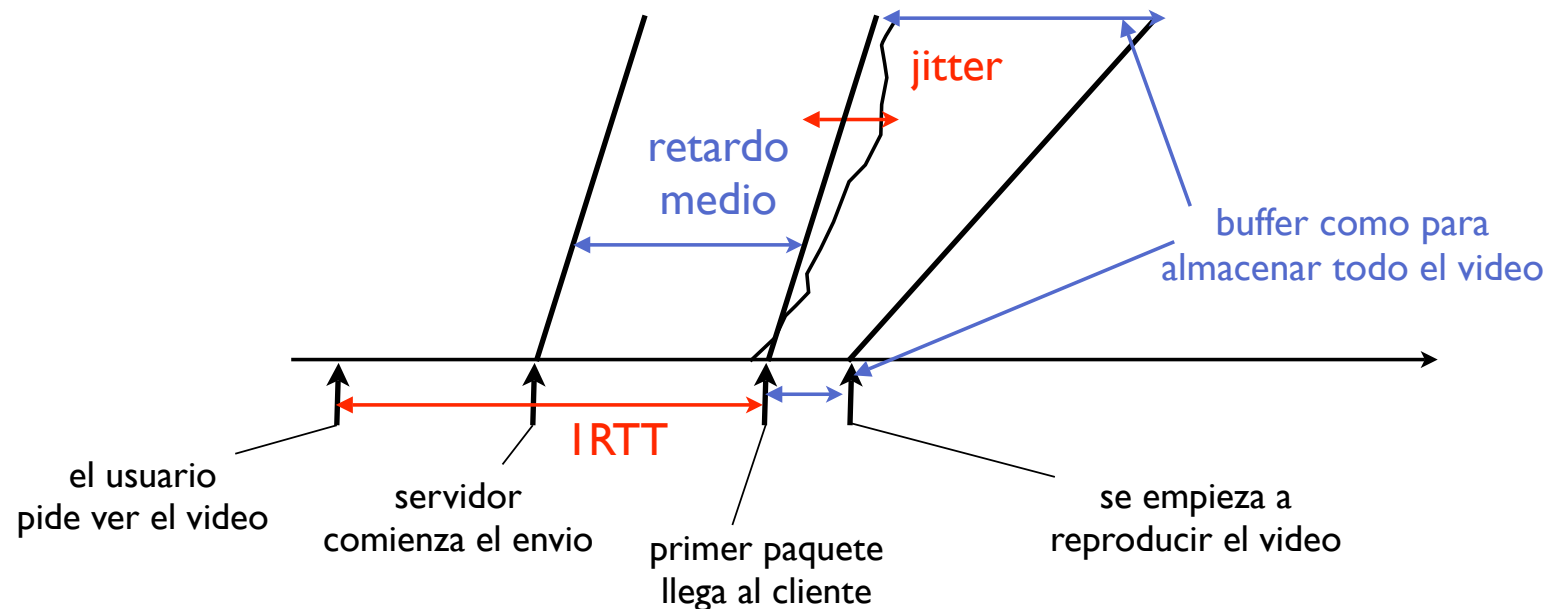
Requisitos de los servicios

- ▶ Streaming de fichero grabado (película)
 - BW suficiente para acomodar la tasa de la película
 - El tiempo de retardo one-way no importa demasiado
 - contribuye al retardo inicial del comienzo de visualización
 - El jitter condiciona el tamaño del buffer
 - y aumenta el retardo inicial del comienzo de visualización
 - si el jitter es mayor que el buffer el video puede detenerse
 - El desorden se compensa con el buffer
 - Perdidas causan cortes



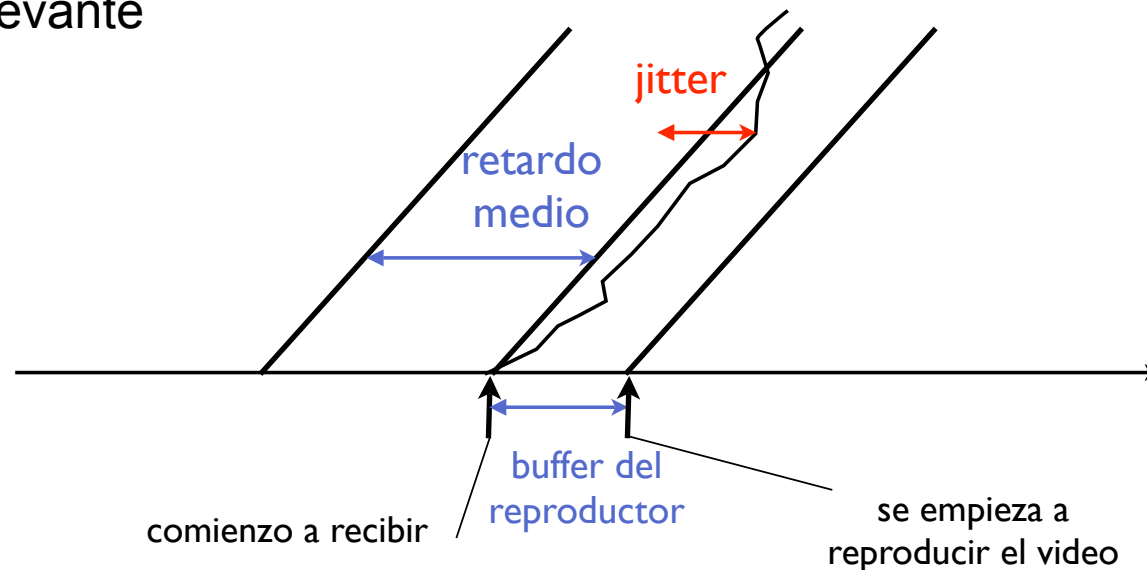
Requisitos de los servicios

- ▶ Streaming de fichero grabado (película)
 - Enviando a más velocidad que la de reproducción de video
 - disminuimos la importancia del jitter... como el video llega mas rapido que se reproduce si no se para al principio probablemente no parara
 - necesita en general mas buffer (un recurso)
 - necesita mas velocidad de la red (otro recurso)
 - necesita que el video este disponible con antelación (no siempre es posible)



Requisitos de los servicios

- ▶ Streaming en vivo (no podemos tener el video a mas velocidad de la que se genera)
O un canal de TV programación lineal (no podemos tener buffer para todo el video)
 - BW suficiente para acomodar la tasa de la pelicula
 - Retardo irrelevante
 - Desorden irrelevante



Notas sobre QoS para streaming

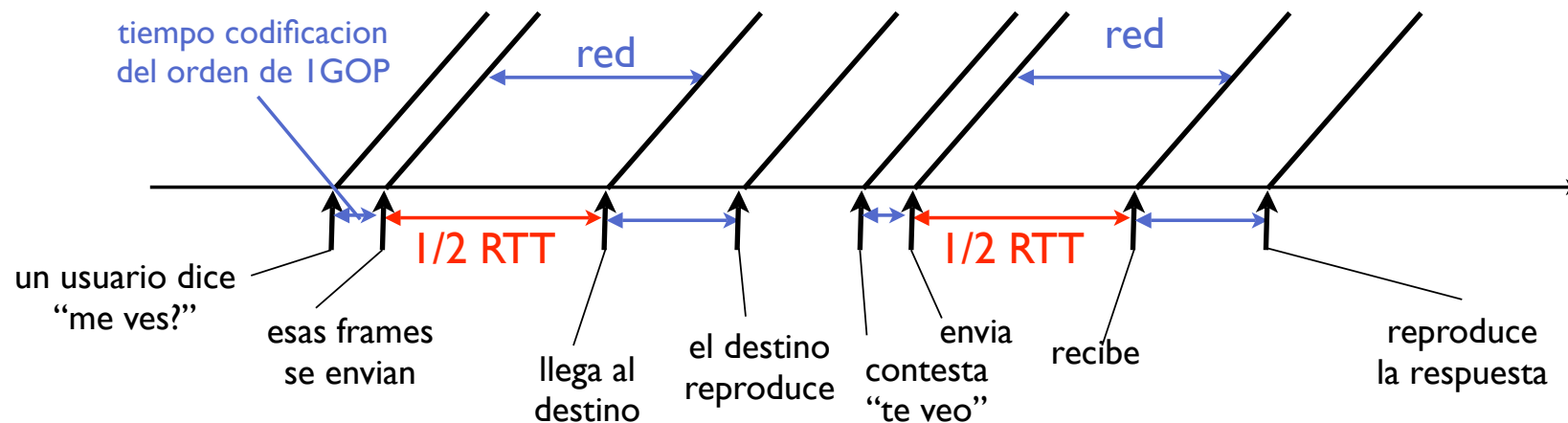
- ▶ Se recomienda marcar como DSCP CS4
- ▶ Perdidas menores del 5%
- ▶ Latencia menor de 4-5 segundos
- ▶ Sin requerimientos de jitter (los compensara el buffer)
- ▶ BW garantizado (CBWFQ) dependiendo de la codificación

- ▶ Flujos unidireccionales en dirección al cliente:
No requiere QoS hacia arriba
- ▶ Otras fuentes de video externas pueden ser marcadas como Scavenger DSCP CS1

Requisitos de los servicios

▶ Video conferencia (2 transmisiones de video)

- BW suficiente
- El retardo aqui si que importa
 - El RTT es inevitable
 - El tiempo máximo aceptable para que un usuario no perciba claramente el delay es de unos 300ms $RTT+2*jitter < 0.3$
 - Tiempo de codificación importa (compromiso con BW)
 - El buffer de reproduccion importa
tiene que ser lo menor posible -> el jitter tiene que ser lo menor posible



Notas sobre QoS para videoconf

- ▶ Recomendado marcar como DSCP AF41
Tráfico en exceso de videoconferencia como AF42 o AF43
- ▶ Pérdidas menores de 1%
- ▶ Latencia one-way menor de 150ms
- ▶ Jitter menor de 30ms
- ▶ Asignar a segunda cola con prioridad (LLQ)
- ▶ Calcular BW para reservar como un 20% más de la tasa de la videoconferencia (i.e. 384kbps reservar 460kbps) para los picos instantáneos tramas I

Más servicios

- ▶ Video bajo demanda con interacción
 - Parar la película
 - Avance rápido
 - Volver atrás
 - RTT y tiempos de buffer dan la velocidad de interacción
 - Se hace normalmente con unicast a cada cliente. Dimensionado de número de clientes

- ▶ Video casi bajo demanda
 - ¿Cuántos de nuestros clientes están viendo a la vez la misma película?

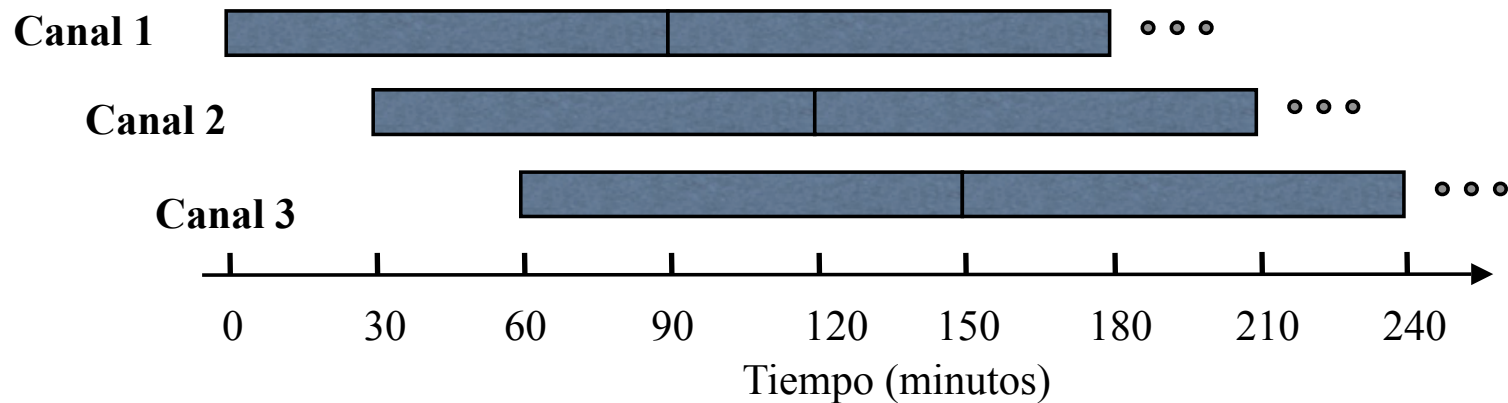
Streaming escalable: Motivación

- ▶ Consideremos un fichero popular
 - Tasa de reproducción (*playback rate*): 1 Mbps
 - Duración: 90 minutos
 - Peticiones: una por minuto
- ▶ Comienza un nuevo stream por cada petición
 - BW requerido = 1 Mbps x 90
- ▶ ¿Cómo dar flexibilidad en el instante de comienzo sin enviar un flujo por petición?
- **Batching**
 - Acumular suficientes peticiones para que sea rentable iniciar un nuevo flujo
 - Empezar el flujo al acumular suficientes usuarios o llegar a un límite de tiempo de espera



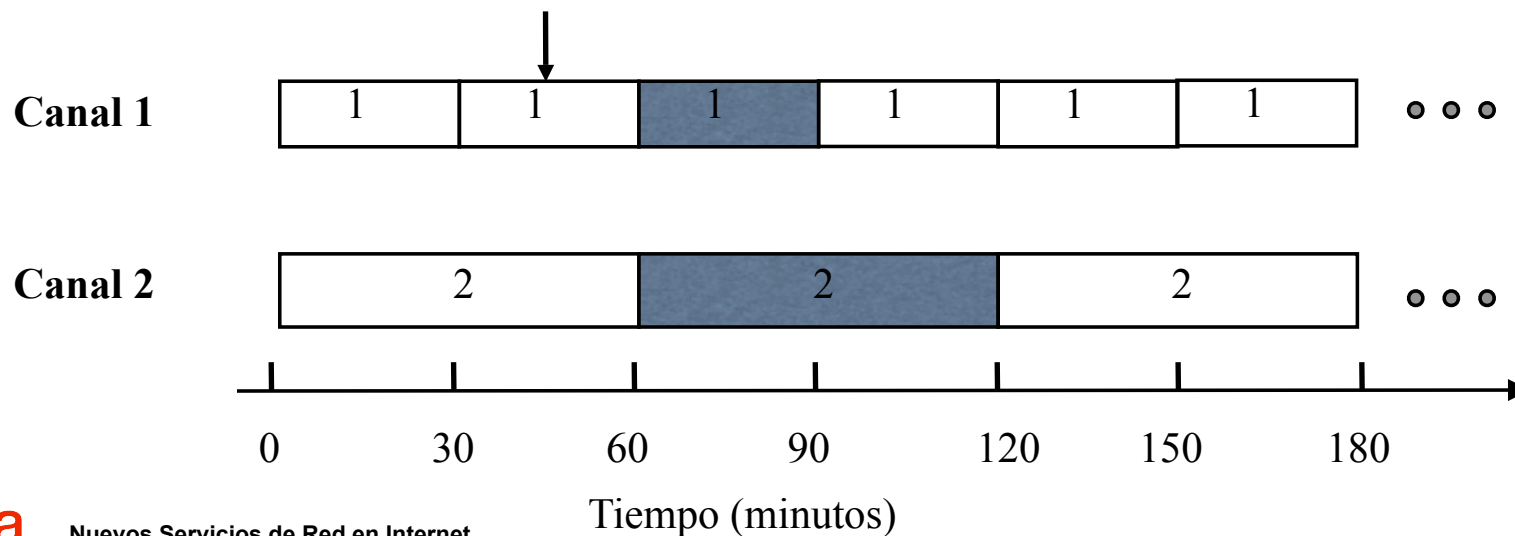
Batching (Ejemplo)

- ▶ Playback rate = 1 Mbps, duración = 90 minutos
- ▶ Agrupar peticiones en intervalos no solapados de 30 minutos:
 - Máx espera comienzo = 30 minutos
 - BW necesario = 3 canales = 3 Mbps
- ▶ BW aumenta linealmente con la reducción de la espera de comienzo



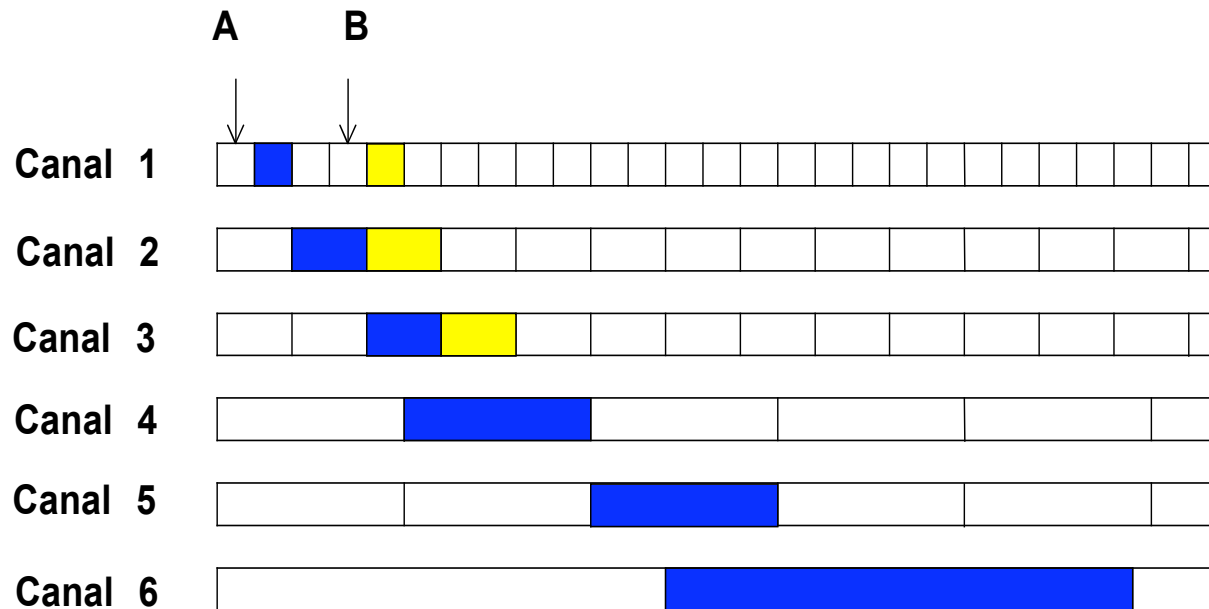
Periodic Broadcast

- ▶ Partir el fichero en 2 segmentos con tamaños relativos {1,2}
- ▶ Para una película de 90 minutos:
 - Segmento 1 = 30 minutos, Segmento 2 = 60 minutos
- ▶ Ventaja:
 - Máx espera comienzo = 30 minutos
 - BW necesario = 2 canales = 2 Mbps
- ▶ Desventajas: Requiere más del cliente (recibir 2 a la vez)



Skyscraper Broadcasts (SB)

- ▶ Divide el fichero en K segmentos de tamaños crecientes
 - Progresión de tamaños: 1, 2, 2, 5, 5, 12, 12, 25, ...
- ▶ Multicast de cada segmento en un canal separado
- ▶ Tasa a cada cliente: 2 x playback rate



Protocolos

Streaming sobre Internet : ¿UDP o TCP?

UDP

- ▶ El servidor envía a la velocidad apropiada para la reproducción
- ▶ Retardo de comienzo (2-5 segs) para compensar el jitter de la red
- ▶ Recuperación de pérdidas: *time permitting*

TCP

- ▶ Puedes entregar a la velocidad de reproducción pero no sabes cuándo lo enviará TCP
- ▶ Además puede acumular bytes para formar paquetes más grandes
- ▶ Normalmente envía a la máxima velocidad
- ▶ Fluctuará debido a los mecanismos de control de flujo y control de congestión
- ▶ Retardo de comienzo más largo para poder suavizar el comportamiento de TCP
- ▶ HTTP/TCP atraviesa mejor firewalls

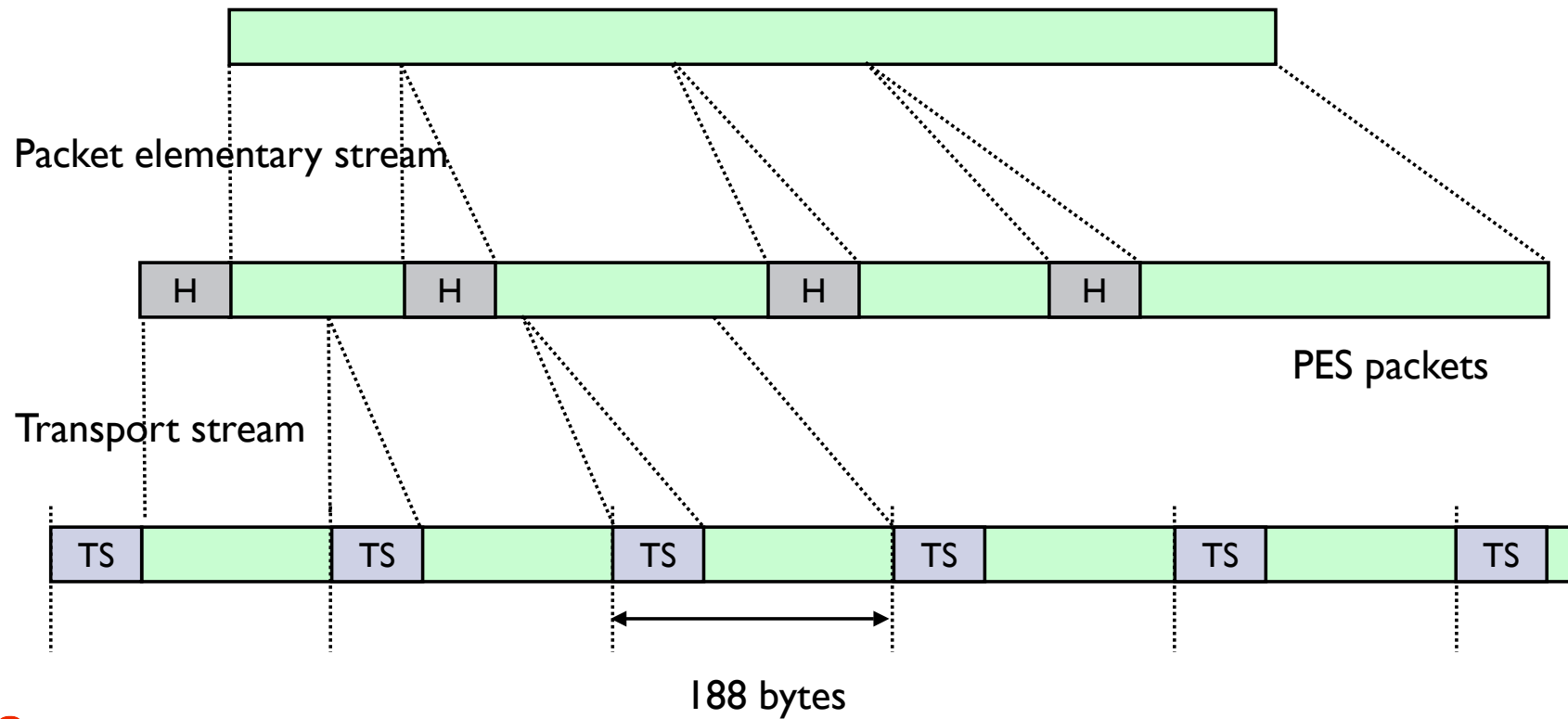
Multimedia en la Internet actual

- ▶ TCP/UDP/IP: “best-effort service”
- ▶ Sin garantías de retardo o pérdidas
- ▶ Los programas emplean técnicas en el nivel de aplicación para mitigar los efectos de retardos y pérdidas
 - Buffers
 - Algoritmos de codificación resistentes ante pérdidas

Distribución de canales

- ▶ MPEG transport stream
MPEG2 part 1

Elementary stream de audio o video



Arquitecturas

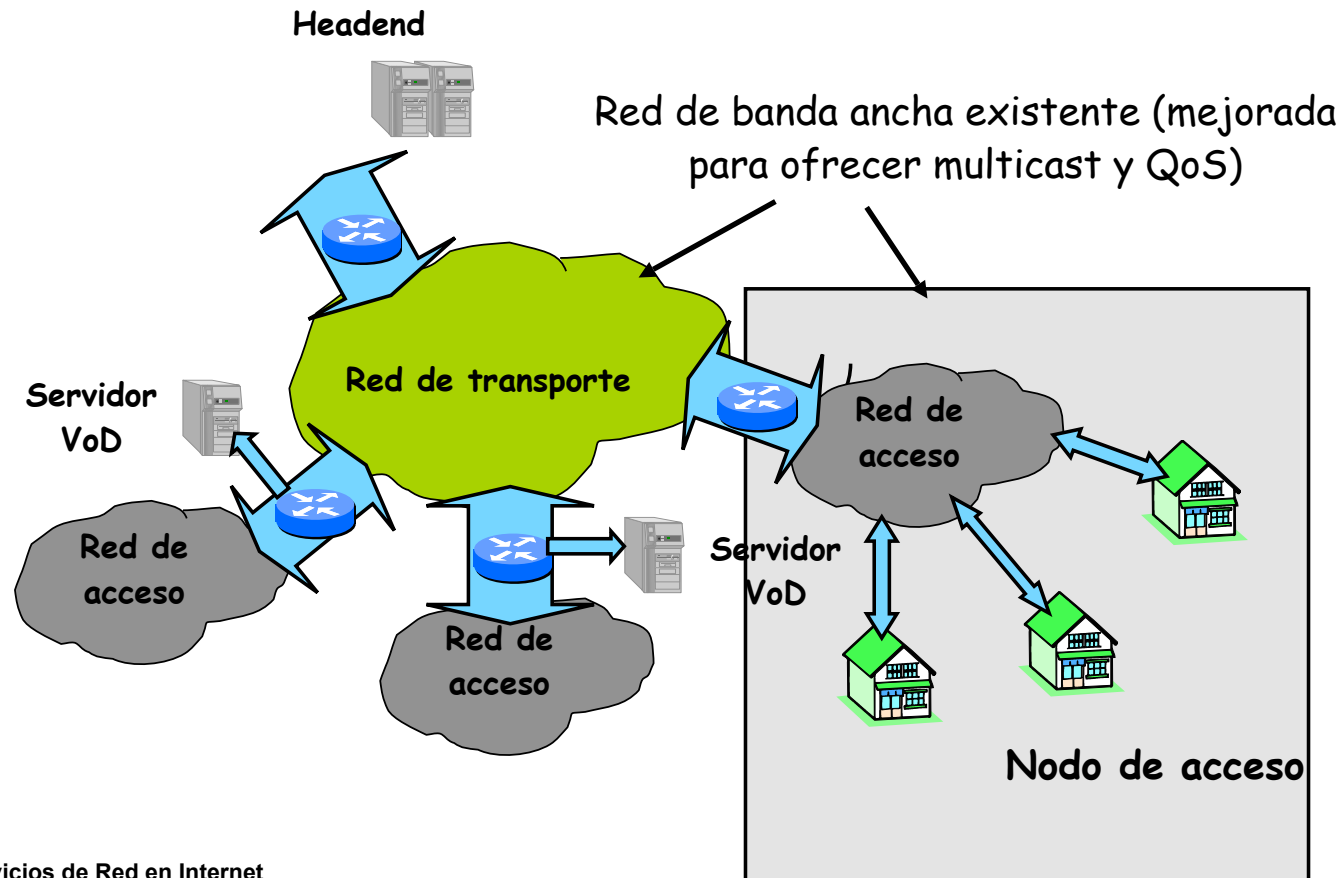
Necesidades de la red

- ▶ BW
- ▶ Retardo (velocidad de interacción)
- ▶ Jitter (frecuencia de cortes o tamaño del buffer)
- ▶ Perdidas (frecuencia de cortes)

- ▶ Situación actual
 - Video sobre internet. Sobredimensionar BW y esperar que los otros tres no sean demasiados, sin garantías
 - Video en entornos controlados por un operador
 - IPTV
 - VoD
 - Mezclas, batching

Arquitectura general

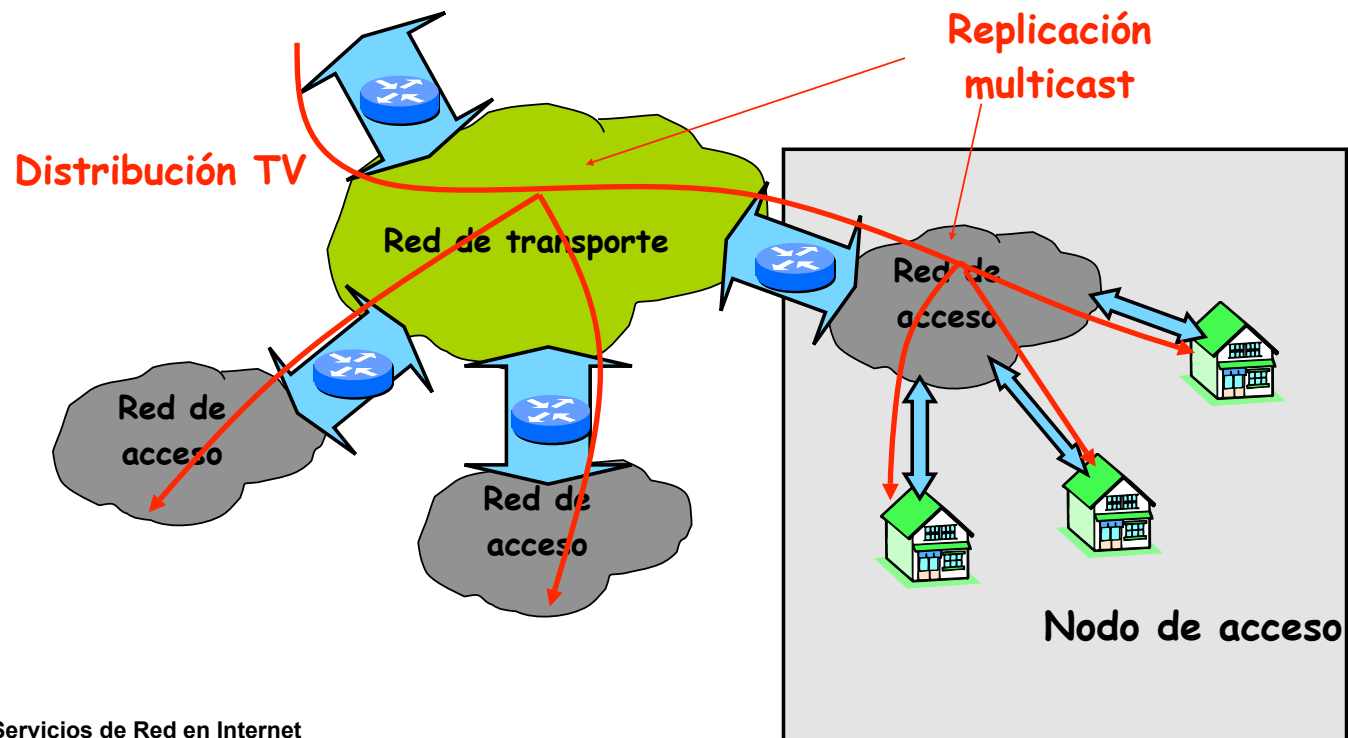
(¿ Imagenio ?)



Difusión de TV

Multicast para difusión de TV

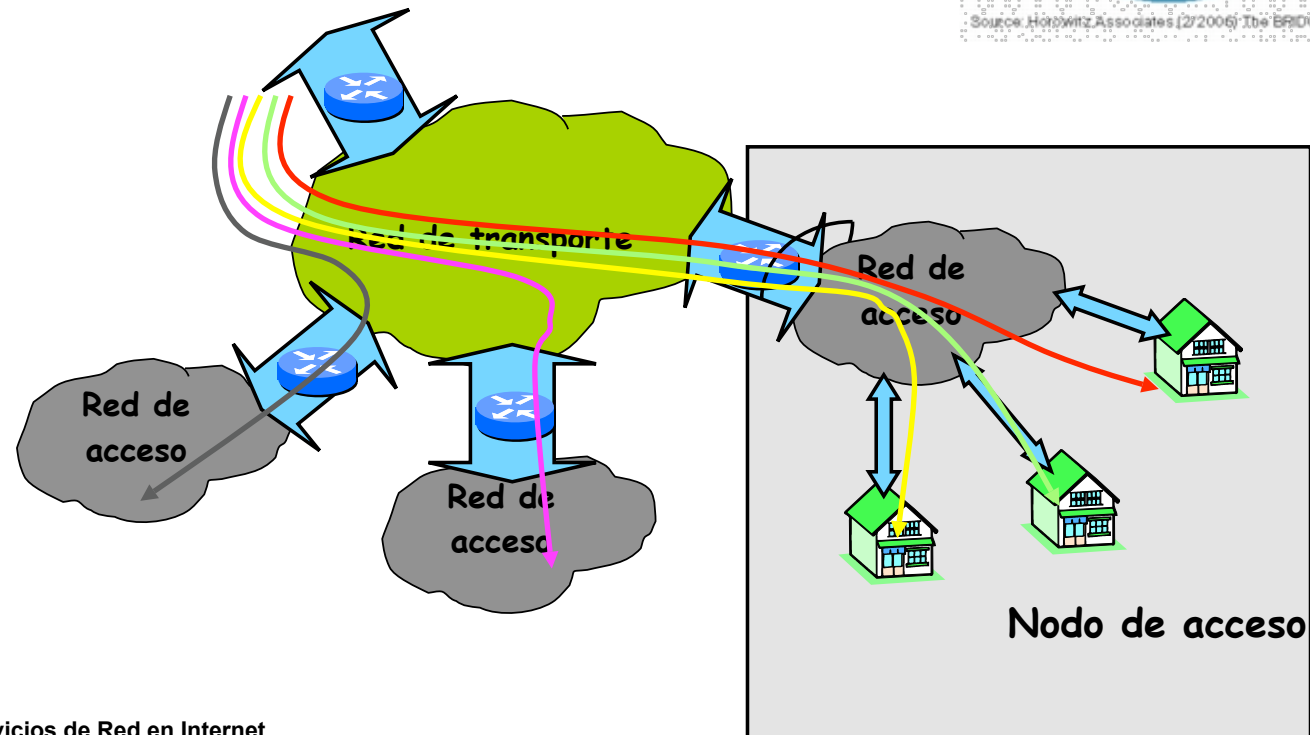
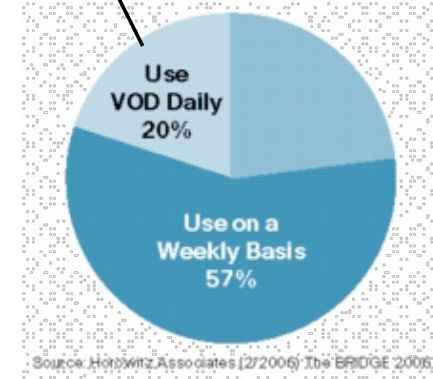
- ▶ Escalabilidad para un alto número de canales y un número ilimitado de usuarios
- ▶ Uso óptimo del ancho de banda



VoD

VoD equivale a Unicast

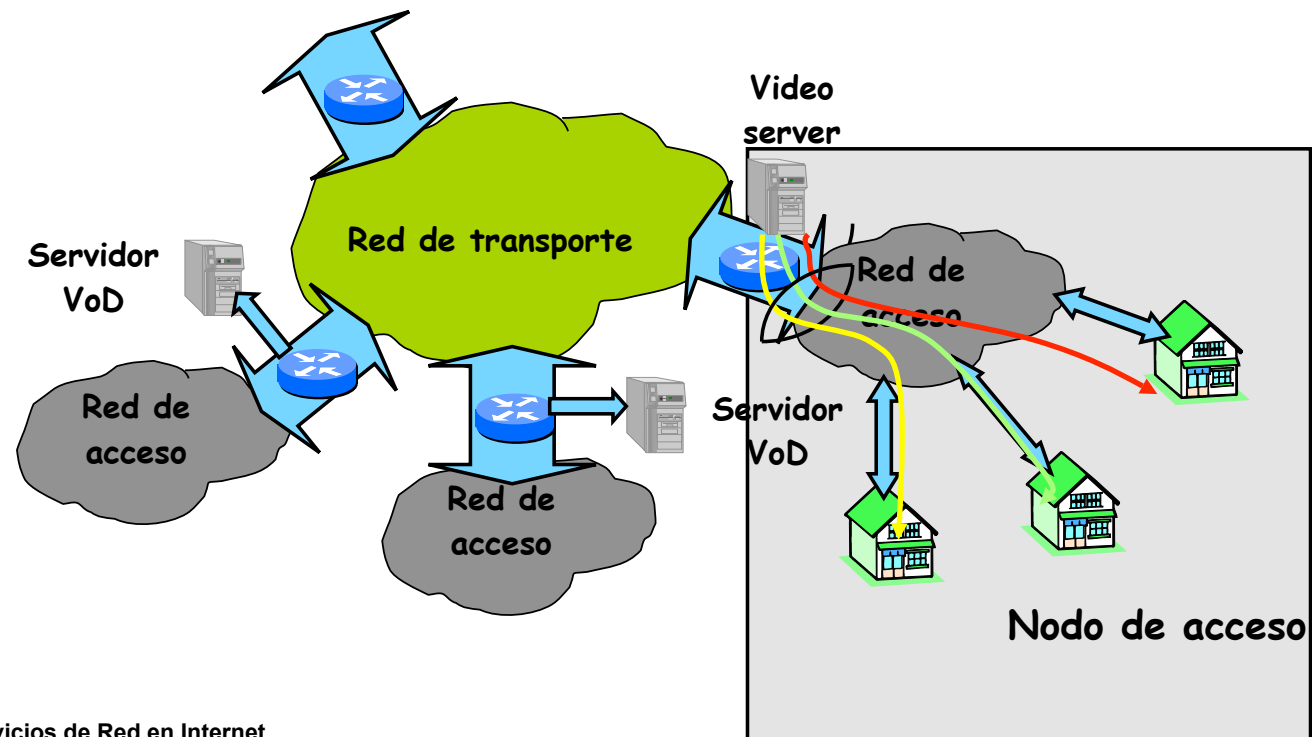
- ▶ 1M usuarios → 200K/día @ 2Mb/s & 20% concurrencia = 80Gb/s
- ▶ Arquitectura centralizada proporciona escalabilidad limitada



VoD

Arquitectura distribuida de VoD

- ▶ 20k usuarios @ 2Mb/s & 20% concurrencia = 8Gb/s
- ▶ Arquitectura distribuida proporciona escalabilidad
- ▶ También es posible una arquitectura mixta (centralizada/distribuida)



Video en el acceso

- ▶ Distribución MPEG transport stream
 - HFC
 - ADSL
 - FTTH
- ▶ Sobre canal de datos IP

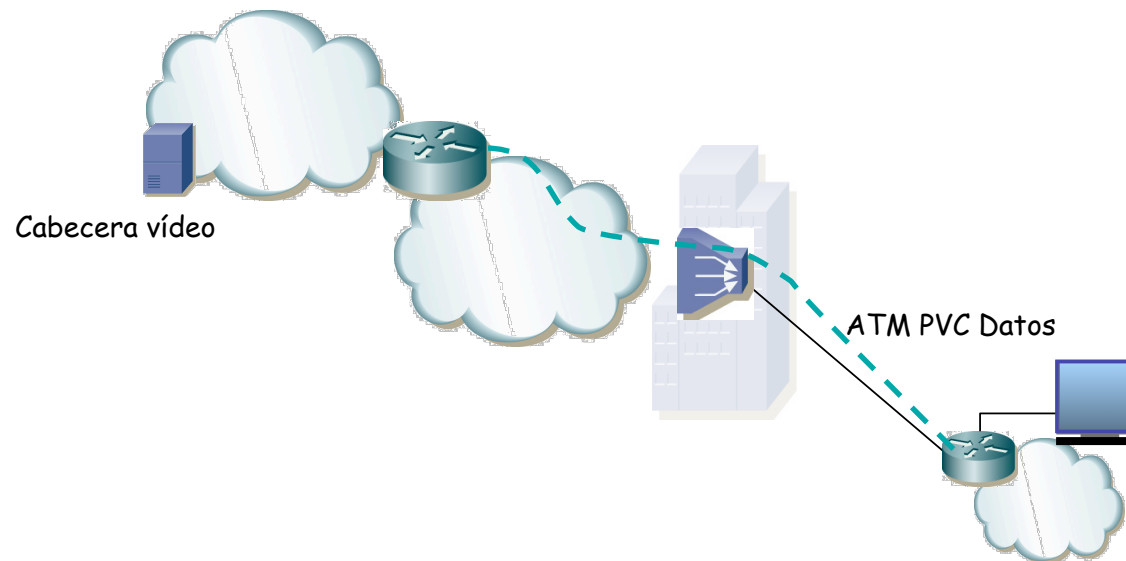
Video en la red de transporte

- ▶ Distribución MPEG transport stream
 - MPLS?
 - ATM

- ▶ Transporte para el video por canales independientes
 - GbE
 - ATM OC-192
 - WDM
 - ROADM

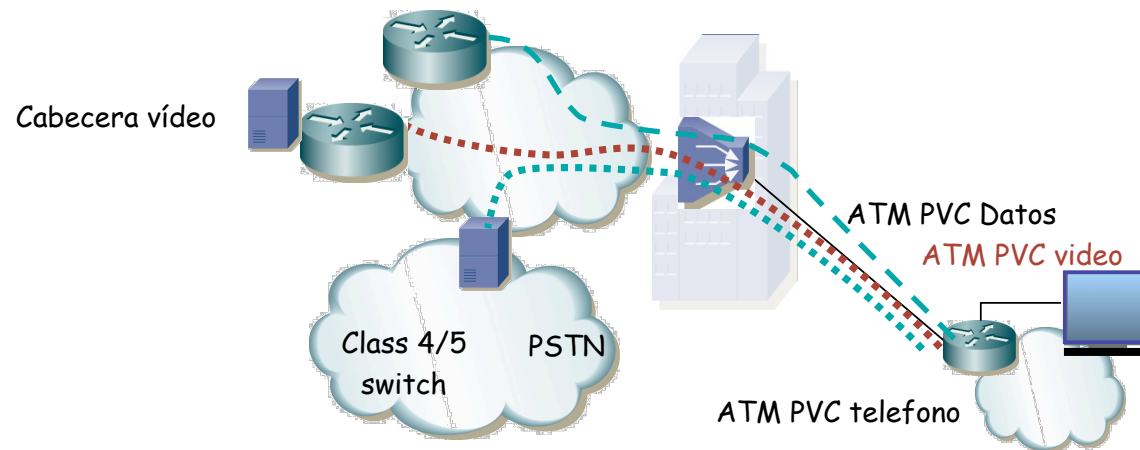
Posibilidades

- ▶ Acceso ADSL
- ▶ Un solo PVC para datos
 - Transporte de 1 canal de vídeo sobre ATM
 - Transporte de 1 canal de video sobre red IP del operador
- ▶ El acceso puede ser ADSL o HFC sobre los datos



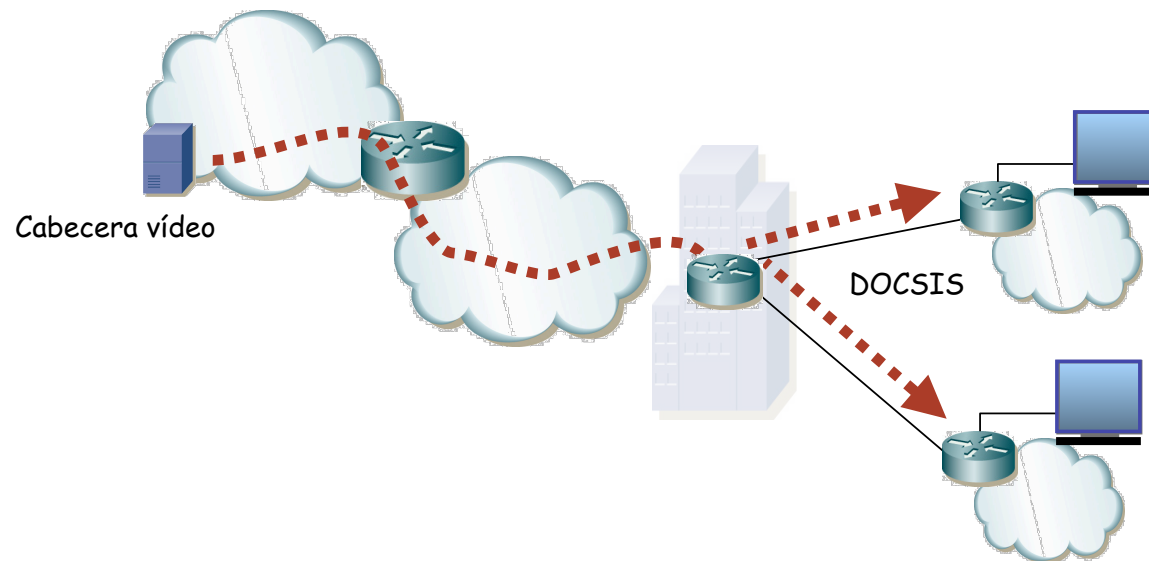
Posibilidades

- ▶ Acceso ADSL +1CV para el video
- ▶ Transporte de 1 canal de vídeo sobre ATM
 - Se puede enviar el canal a mas usuarios con multipunto ATM



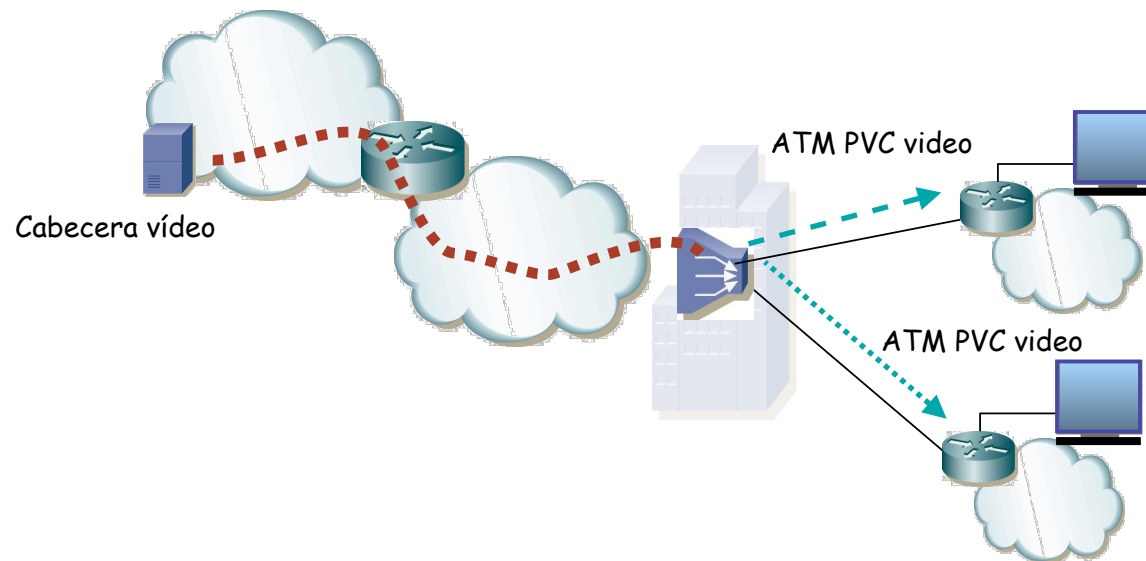
Posibilidades

- ▶ Acceso HFC
- ▶ Varios canales de video se distribuyen sobre DOCSIS
 - Transporte de TS MPEG sobre HFC/DOCSIS
 - Transporte de TS MPEG en la red de la operadora



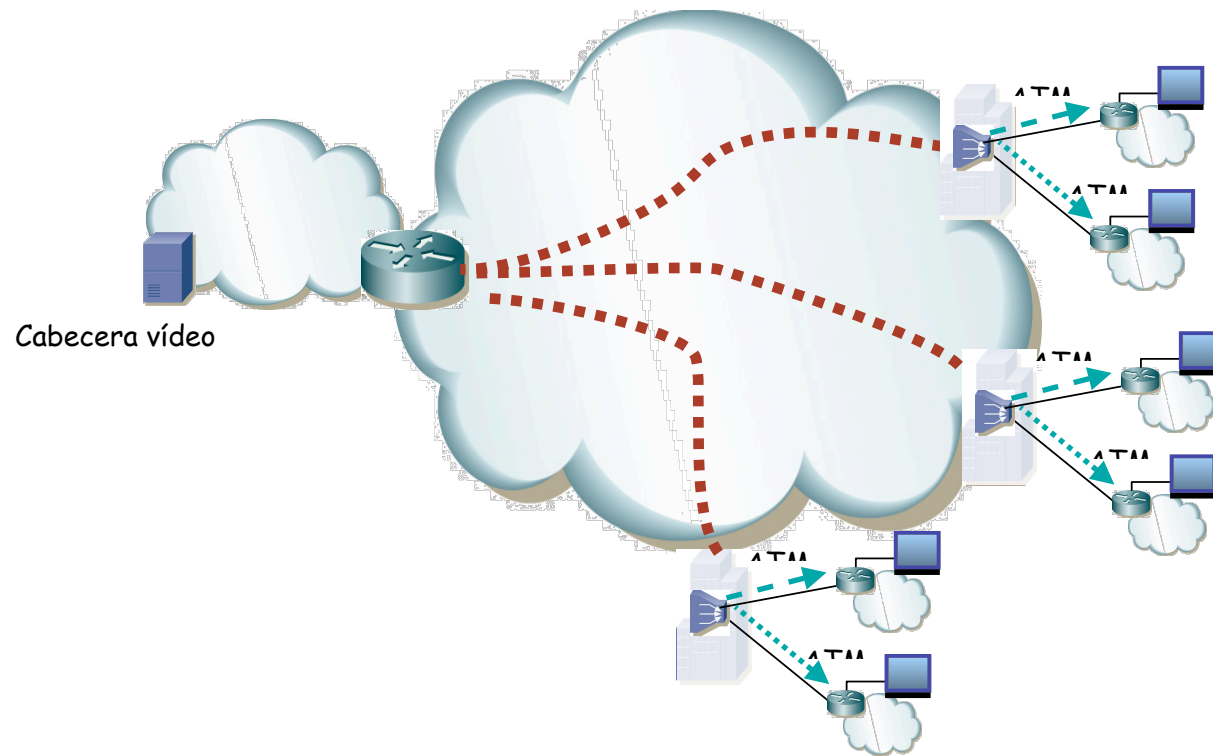
Posibilidades

- ▶ Transporte de TS MPEG en la red de la operadora
 - sobre ATM oc-192 9Gbps
 - sobre infraestructura de 10GbE
 - sobre WDM en las mismas fibras
 - ...



Posibilidades

- ▶ Transporte de TS MPEG en la red de la operadora
 - Envío de una cabecera a multiples servidores en los DSLAMs
 - Punto a multipunto sobre ATM (multicast ATM)
 - Multicast sobre IP
 - Unicast sobre WDM



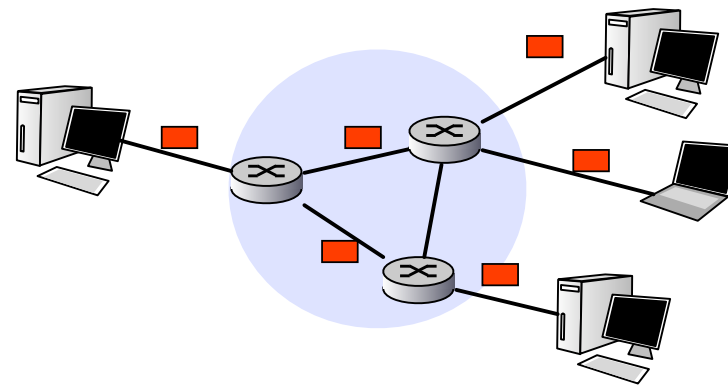
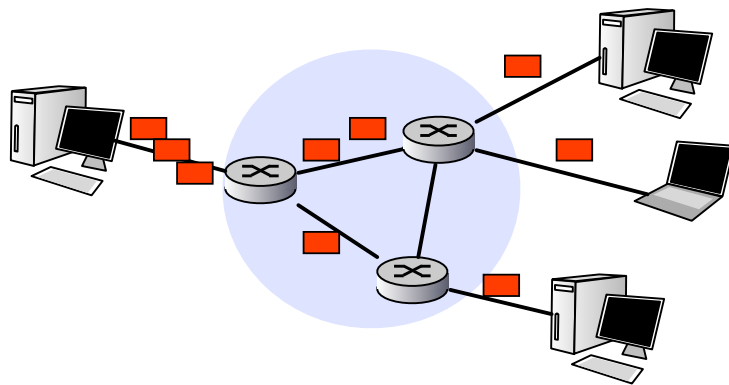
Conclusiones

- ▶ Video y codificación (MPEGs, H62x, GoPs, CBR, VBR)
- ▶ Servicios
 - Play-as-you-get, streaming, VoD, IPTV, Batching, periodic broadcast
 - Requisitos: BW, delay, jitter, buffer de reproducción
- ▶ Arquitecturas
 - IPTV, VoD
- ▶ Necesidades de la red
 - Sin cortes ⇒ QoS
 - Escalabilidad ⇒ **Multicast**
 - **Tecnologías de transporte y equipos**

Multicast en IP

Broadcast y Multicast

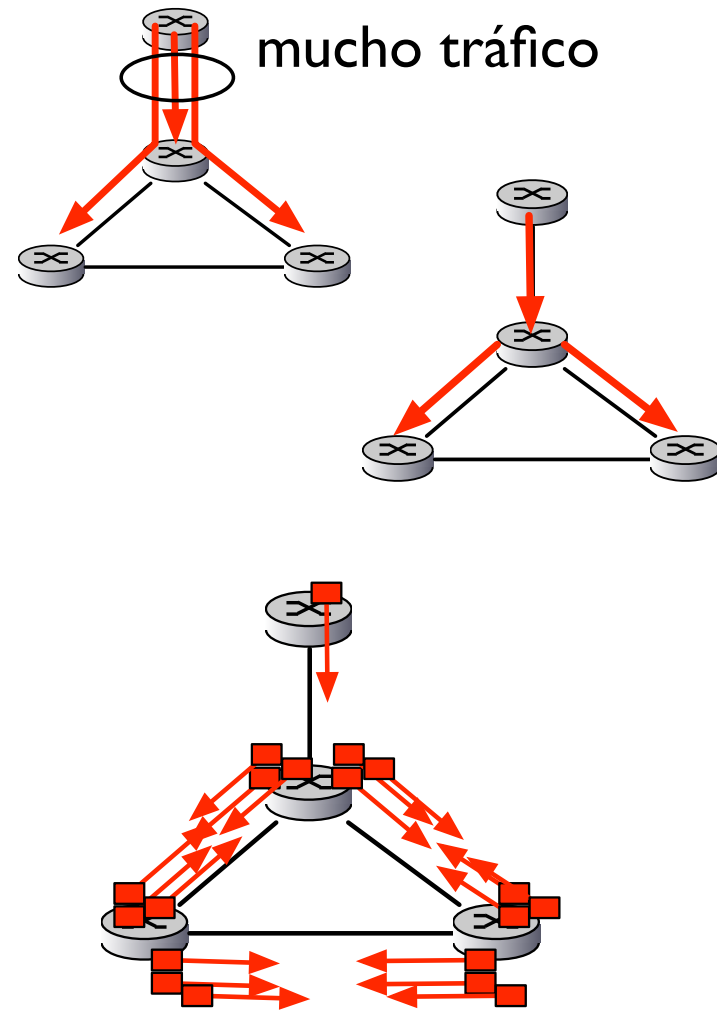
- ▶ Envío de paquetes a un solo destino **Unicast**
- ▶ Envío de paquetes a varios destinos
Todos los destinos (de una red): **Broadcast**
Múltiples destinos: **Multicast**



Broadcast

Broadcast: problemas

- ▶ Duplicación en el origen
 - Demasiado ineficiente
 - No conocemos las direcciones de todos los destinos
- ▶ Duplicación en la red
 - Como?
Inundación (flooding) cada router reenvía los paquetes que le llegan por todos sus enlaces menos por el que ha venido?
Que problemas tiene esto?



Broadcast: métodos

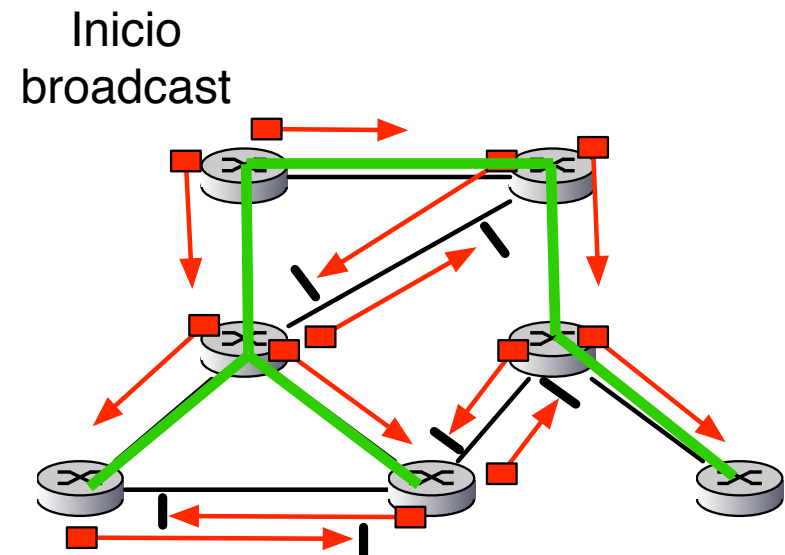
▶ Inundación controlada

para evitar las tormentas de broadcast causadas por los bucles

- Número de secuencia y origen: si los routers recuerdan la dirección origen y el número de secuencia y solo envían una vez cada broadcast

▶ Reverse Path Forwarding

- reenvía solo los paquetes que llegan por el camino más corto hacia el origen
- No necesita estado en el router (sólo tabla de rutas)
- Genera tráfico extra pero no excesivo
- Broadcast sigue el spanning tree



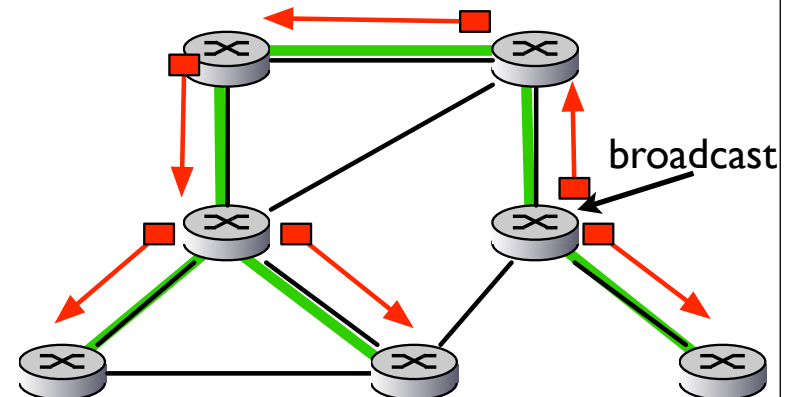
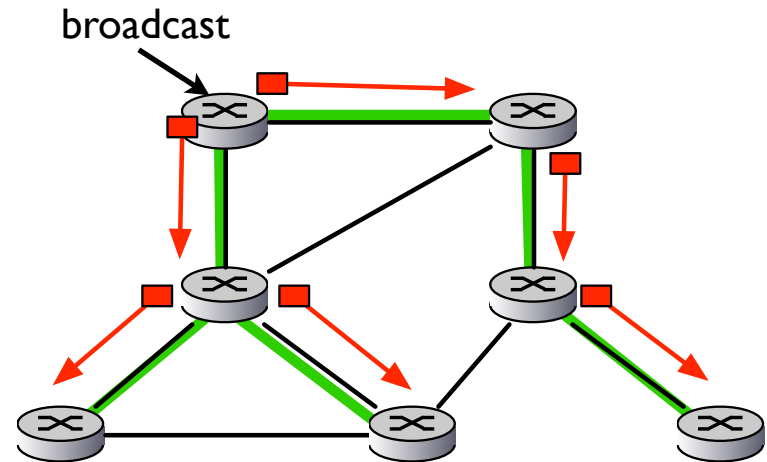
Broadcast: spanning-tree

► Broadcast siguiendo un spanning-tree

- Primer paso construir un spanning tree (centrado en el nodo que inicia el broadcast?)
- Reenviar sólo siguiendo enlaces del spanning tree

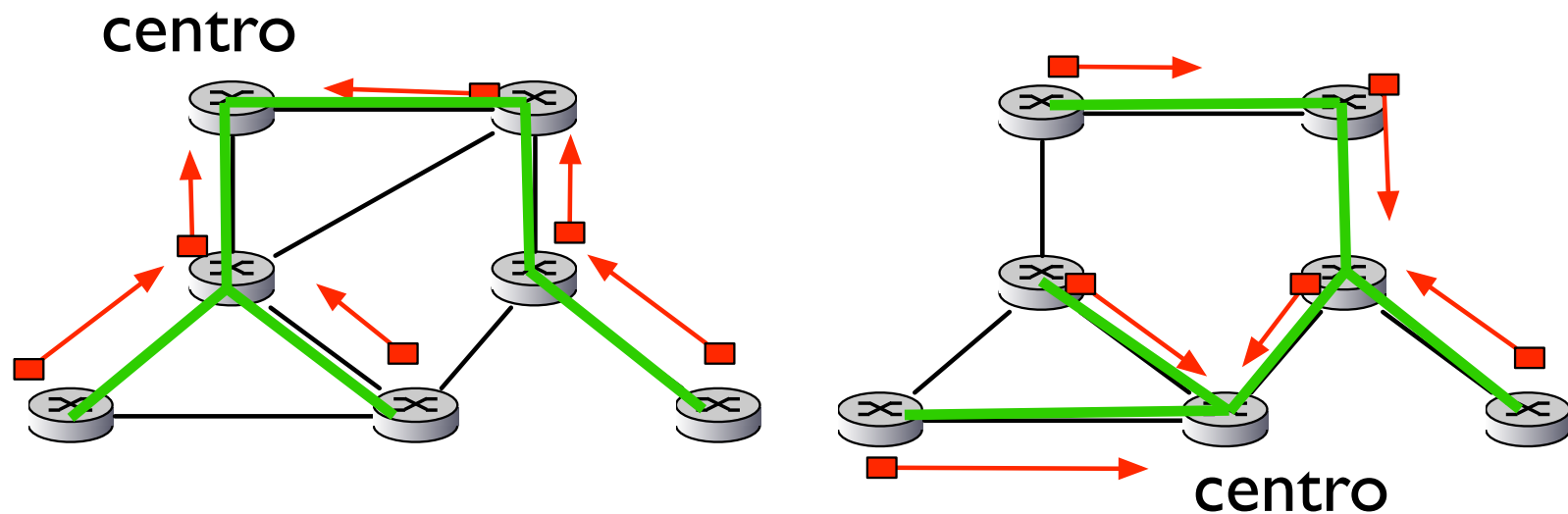
► Ventajas

- Una vez calculado el spanning tree se puede usar cualquier nodo para un broadcast
- Envía solo a tus vecinos en el spanning tree
- Cada nodo no necesita conocer todo el spanning tree, sólo quienes son sus vecinos



Algoritmos de construcción de spanning-tree

- ▶ Basado en un nodo central (rendezvous point o core)
 - Cada nodo envía un paquete al punto central
 - Los caminos que siguen los paquetes forman el spanning tree
- ▶ Sabiendo el centro del spanning tree cualquier router sabe cual es el enlace por el que le tienen que llegar paquetes del spanning tree (el enlace de la ruta mas corta hacia el origen/centro)



Broadcast: aplicaciones

- ▶ Nivel de aplicación: overlay networks.
Ejemplo: Gnutella hace búsquedas por inundación
- ▶ Nivel de red: enrutamiento link-state, el estado de los enlaces debe enviarse a todos los nodos de la red.
Ejemplo: OSPF

IGMP

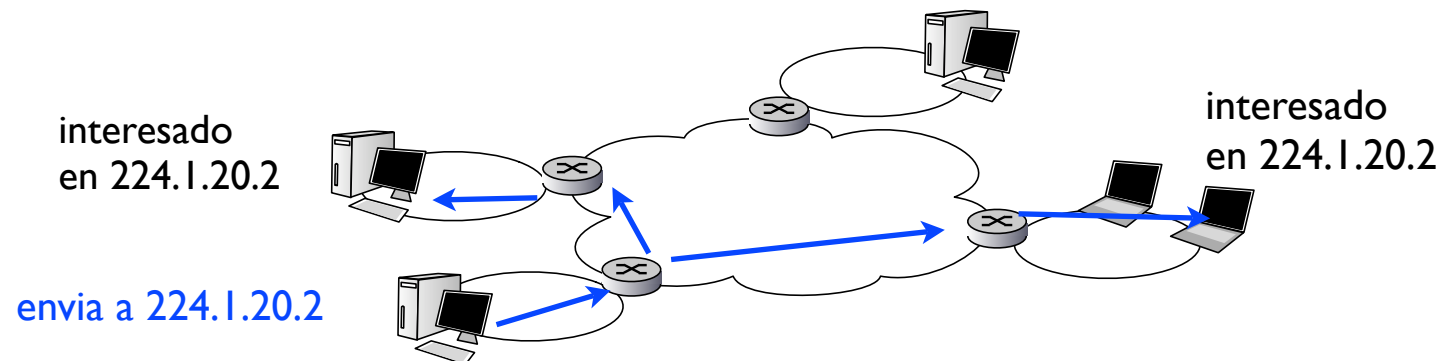
Multicast

▶ Grupos multicast

- No tiene sentido que el paquete lleve la dirección de todos los interesados
- Se usa un grupo multicast identificado por una dirección independiente de la localización
direcciones IP de clase D [rango 224.0.0.0 a 239.255.255.255]

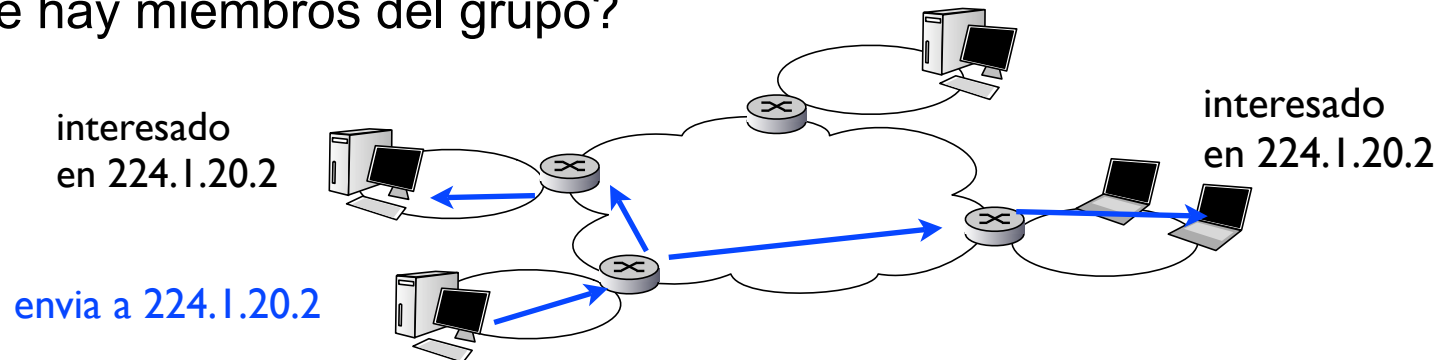


- ▶ El problema es como hacer que llegue un paquete dirigido a una dirección multicast a todos los interesados
 - Sencillo en una LAN
 - No tan sencillo atravesando redes



Multicast

- ▶ La pertenencia a un grupo multicast es dinámica
 - Los interfaces de red pueden suscribirse o no al grupo.
 - Un interfaz puede suscribirse a varios grupos
 - No hay límites de localización o número de miembros
 - No hace falta ser miembro para enviar paquetes al grupo. El grupo son los interesados en escuchar los paquetes que van a una determinada dirección multicast
- ▶ Problemas a resolver
 - Como enviar a varios destinos en una LAN?
 - Como gestionar la pertenencia a un grupo multicast?
 - Como hacer que los paquetes enviados lleguen hasta las LANs en las que hay miembros del grupo?



Grupos multicast con significados especiales

- ▶ Multicast local [224.0.0.0 - 224.0.0.255]

Los routers no los reenvían

Uso en ciertos protocolos que necesitan mandar multicast a la LAN únicamente

- 224.0.0.1 All hosts todos los hosts multicast de la LAN
- 224.0.0.2 All multicast routers todos los routers multicast
- 224.0.0.4 Todos los routers DVMRP
- 224.0.0.5 Todos los routers OSPF
- ...

- ▶ Direcciones reservadas para aplicaciones [224.0.1.xxx]

Si se reenvían por los routers

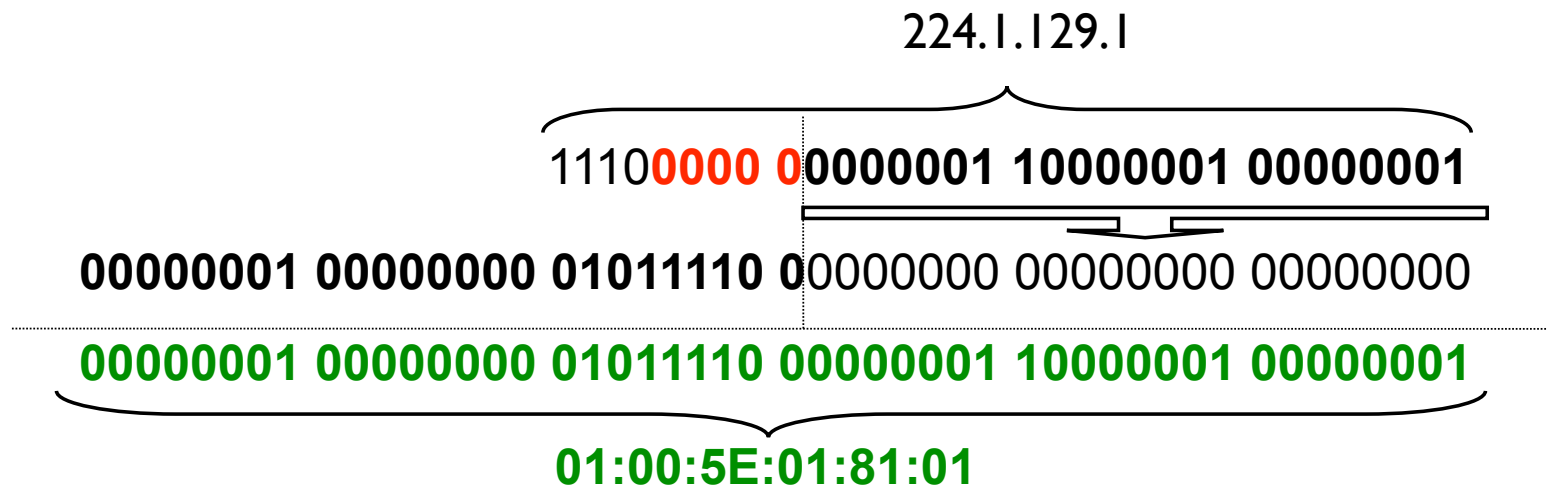
- 224.0.1.1 NTP
- 224.0.1.32 mtrace
- 224.0.1.33-34 RSVP
- 224.0.1.39-40 Cisco RP discovery

- ▶ Multicast privado [239.0.0.0 239.255.255.255]

No se reenvían fuera de una entidad

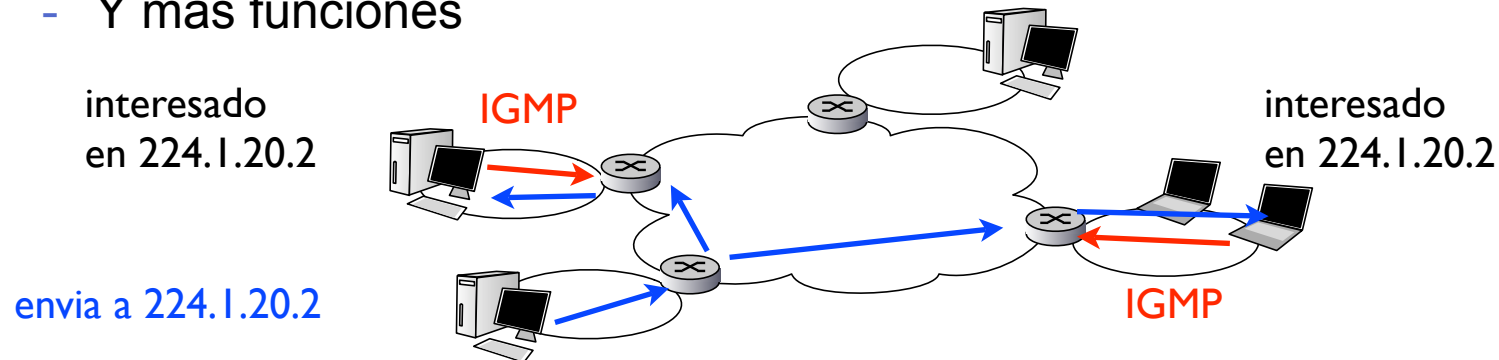
Multicast en LAN Ethernet

- ▶ Caso: Paquete multicast a una LAN con hosts miembros
- ▶ LAN Ethernet soporta multidifusión
- ▶ MACs multicast octavo bit activo
- ▶ IANA posee el OID **01:00:5E**
- ▶ Reserva la mitad para multicast
- ▶ De 01:00:5E:00:00:00 a 01:00:5E:7F:FF:FF
- ▶ En los 23 bits bajos se mapean los 23 bits bajos del grupo multicast destino
- ▶ 32 grupos colisionan en la misma MAC multicast
- ▶ Al suscribir un interfaz a un grupo multicast IP recibe los grupos multicast cercanos y tendrá que descartar los que no quiera



Multicast entre LANs

- ▶ Los routers tienen que saber si hay interfaces interesados en la LAN
 - Para reenviar los paquetes recibidos a la red de área local
 - Para hacer llegar los paquetes con destino multicast a todos los interesados
- ▶ Protocolo de gestión de grupos multicast
IGMP [RFC-2236 igmp v2][RFC-3376 igmp v3]
Internet Group Management Protocol
 - Permite a los hosts informar a su router por defecto de que están interesados en un grupo multicast
 - Y más funciones



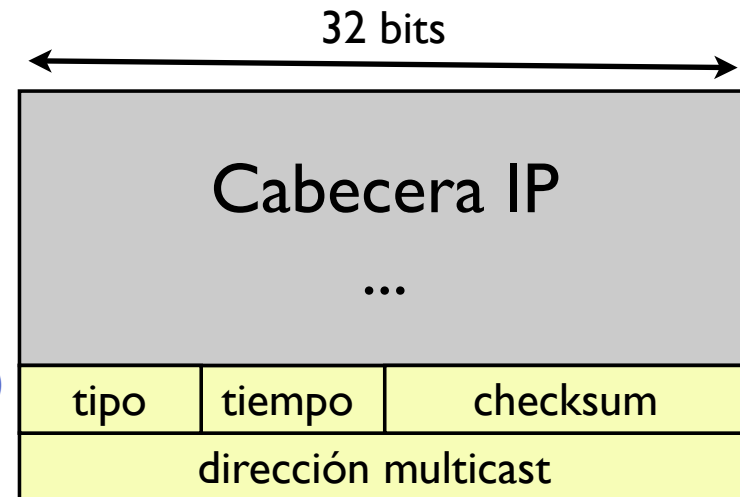
IGMP (v1)

- ▶ Solo 2 mensajes **Query** y **Report**
- ▶ **Mantenimiento**
 - Periódicamente (1min) el router envía Querys a All-hosts (224.0.0.1)
 - Los interesados mandan un Report de los grupos que quieren (tiempo de espera aleatorio 0-10s)
 - Si un interesado ve un Report de otro no se molesta en informar
Eficiente a costa de no mantener estado de todos los interesados en el router
- ▶ **Unirse a un grupo:** Enviar unsolicited report
- ▶ **Dejar un grupo:** Dejar de enviar reports
 - Si un router no recibe reports en 3 intentos (3min) elimina el grupo
- ▶ Problemas:
 - Tiempo de reacción a unirse a un grupo aceptable
 - Tiempo de reacción a dejar un grupo... demasiado lento

¿Que pasa si uso esto en un sistema que envía canales de video en grupos multicast y el usuario hace zapping?

IGMP (v2)

- ▶ Sobre IP (8 bytes)
 - tipo de mensaje
 - tiempo máximo de respuesta (x0.1s) para algunos mensajes
 - checksum (del mensaje IGMP sólo)
 - dirección multicast (en algunos mensajes)
- ▶ 3 tipos de mensajes
 - Membership report (tipo=0x16) informa de interes en un grupo
 - Leave report (tipo=0x17) informa de que deja un grupo
 - Query (tipo=0x11)
 - group specific query pregunta si hay alguien interesado en un grupo
 - general el router pide actualización de los grupos



IGMP(v2)

▶ Unirse a un grupo

- Enviar unsolicited reports indicando el grupo

▶ Mantenerse en un grupo

- El router pide actualización de grupos con tiempo de respuesta indicados [por defecto 10s]
- Los interesados contestan en un tiempo aleatorio [0-indicado]
- Si alguien ya ha pedido el grupo se abstienen

▶ Dejar un grupo

- Enviar un leave report a All-routers 224.0.0.2 indicando el grupo
- El router hace specific-queries del grupo [default 1s y 2 intentos]
si no hay interesados elimina el grupo
- O bien si no se responde a los mensajes de mantenerse eventualmente el router descarta el grupo

▶ Elección de router responsable de los queries

- Cuando un router ve un query de otro router con menor dirección IP le cede la responsabilidad. Si no ve queries en un tiempo lo reintenta
- En IGMPv1 no hay eleccion de responsable, es responsabilidad del protocolo de enrutamiento multicast

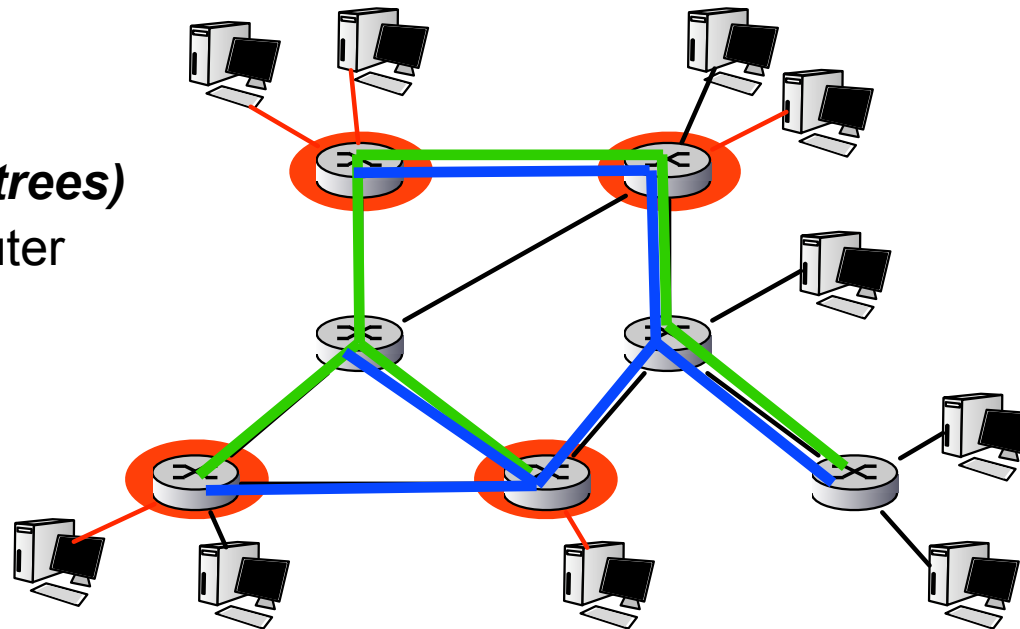
IGMP (v3)

- ▶ Control para escuchar fuentes multicast individuales dentro de un grupo
- ▶ Se puede informar de interes en varios grupos en un solo paquete
- ▶ Envío de reports a 224.0.0.22 All-IGMPv3ENABLED
Permite el multicast-snooping por equipos de nivel 2
- ▶ No hay supresión de informe de interés en un grupo multicast aunque vea a otros interesados, de forma que se puede llevar el estado individual de los interesados
- ▶ Multicast snooping
 - Equipos de nivel 2 que observan los paquetes IGMP provenientes de los puertos para reenviar multicast solo a los que lo solicitan.

Enrutamiento multicast

Enrutamiento multicast

- ▶ Encontrar un árbol (o varios) para enrutar el tráfico multicast entre los routers que tengan hosts interesados en el grupo
- ▶ 2 formas
 - **source-based**
SPT (shortest path trees)
un árbol por cada router
 - **group-shared**
Shared trees
un árbol para todo el grupo multicast



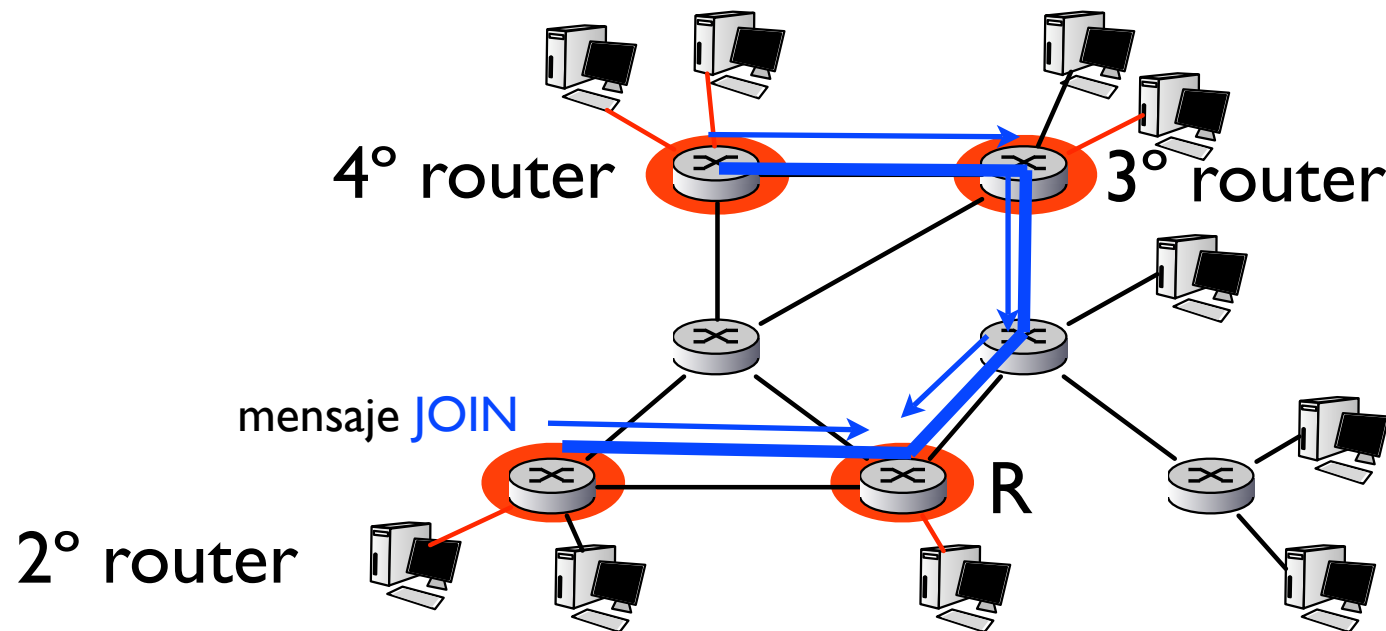
Enrutamiento multicast (group shared)

- ▶ Un unico arbol por grupo multicast
- ▶ Algoritmos
 - Hay algoritmos centralizados para calcular el arbol de minimo coste
Son estáticos y necesitan demasiada información de la red.
 - Algoritmos basados en un nodo central
 - Se identifica un router como centro
 - Los routers que se van uniendo envian mensajes al router central que sirven para construir el spanning-tree a ese nodo

 - El problema es la elección del nodo central
- ▶ Los shared trees se denotan por la dirección del grupo multicast al que pertenecen y un * indicando que es para todas las fuentes
i.e.: shared tree (*,224.1.1.1)

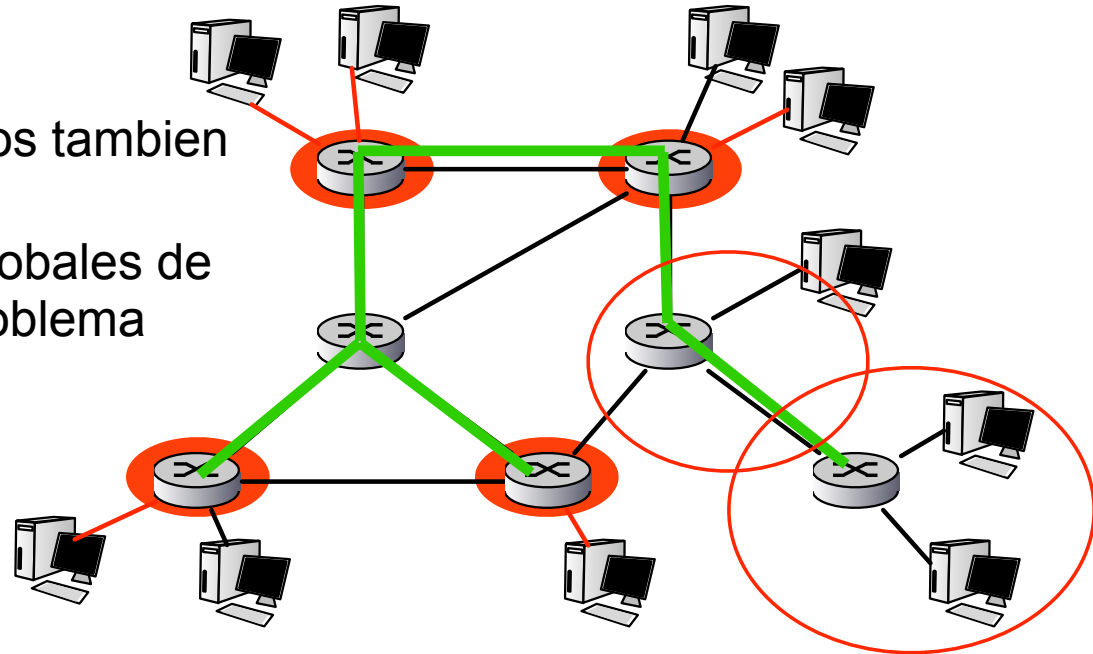
Enrutamiento multicast (group shared)

- ▶ Ejemplo:
El router R se elige como centro.
Al ir añadiendo nuevos routers se forma el spanning-tree



Enrutamiento multicast (source based)

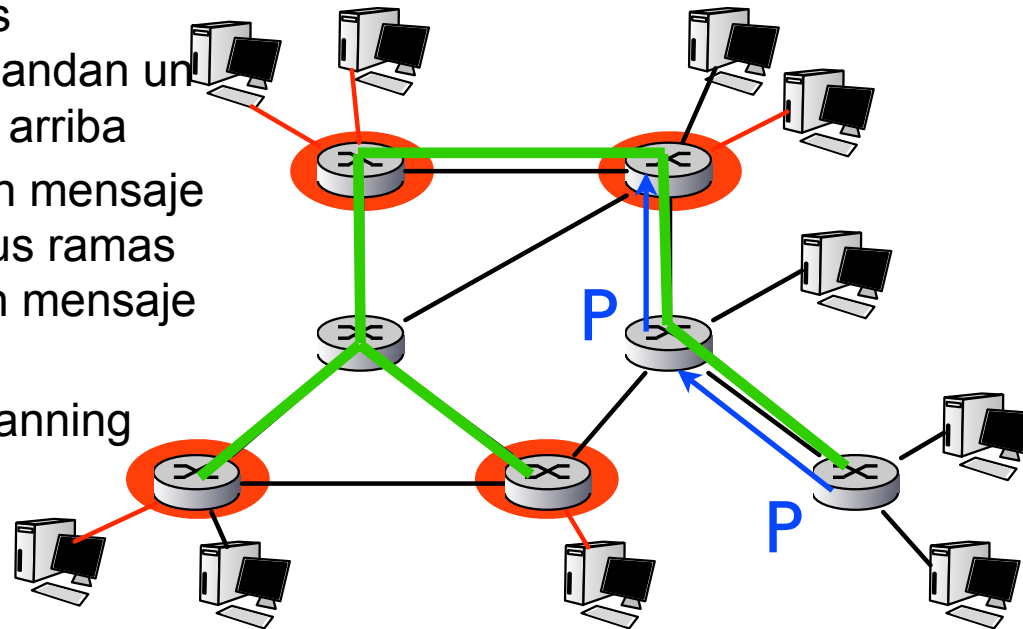
- ▶ Un arbol por cada fuente
- ▶ Usando Reverse Path Forwarding
- ▶ Problema:
 - Los nodos no interesados tambien reciben
 - Los primeros intentos globales de multicast tenían este problema



Enrutamiento multicast (source based)

- ▶ Solución: podar el arbol !!
- ▶ RPF con pruning (poda)

- Los routers que reciben un mensaje multicast y no tienen hosts interesados en el grupo mandan un mensaje de pruning hacia arriba
- Los routers que reciben un mensaje de pruning desde todas sus ramas del spanning tree envían un mensaje de pruning hacia la raíz
- La rama se elimina del spanning tree



Enrutamiento multicast (source based)

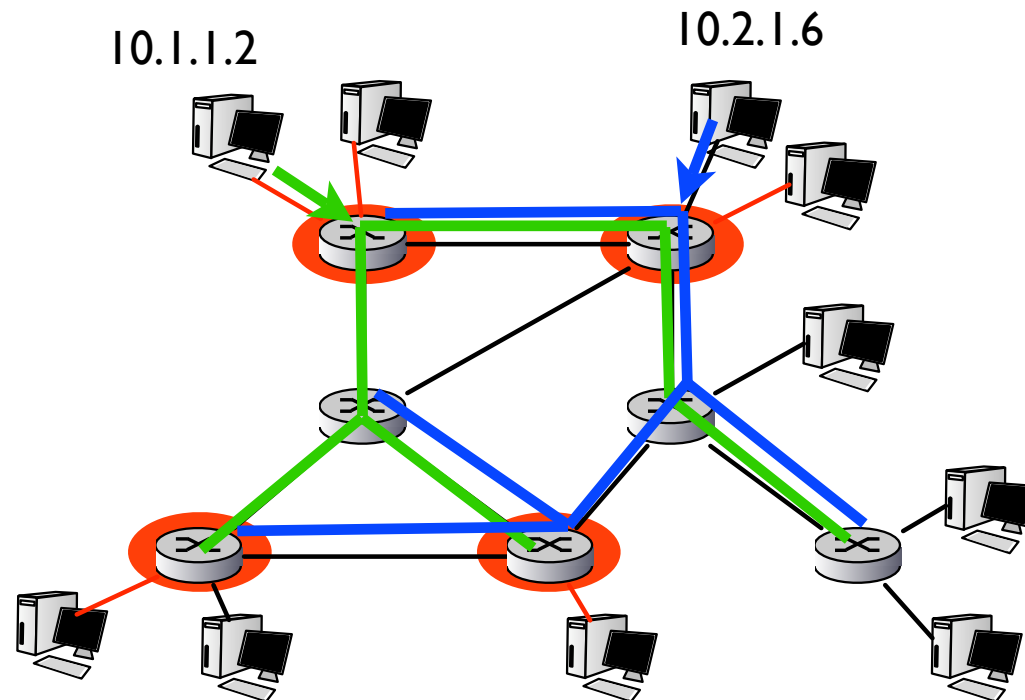
- ▶ Notese que en un mismo grupo multicast cada fuente se distribuye por un SPT diferente
- ▶ Notación para los shortest path trees

SPT1

(10.1.1.2 , 224.1.1.1)

SPT2

(10.2.1.6 , 224.1.1.1)



Protocolos

Protocolos de enrutamiento multicast

- ▶ **DVMRP**

Distance vector multicast routing protocol (RFC-1075)

- ▶ **PIM**

Protocol Independent Multicast routing protocol (RFC-2362)

- ▶ **MOSPF**

Multicast OSPF

- ▶ **CBT**

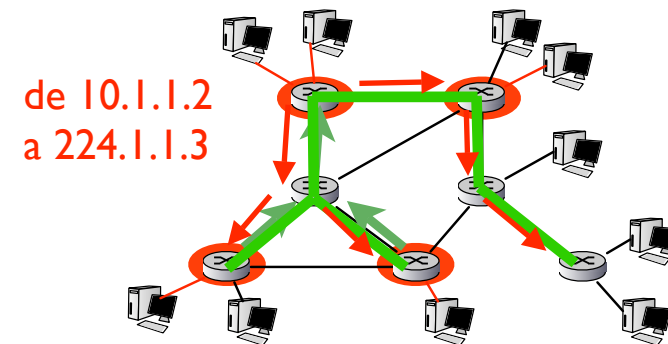
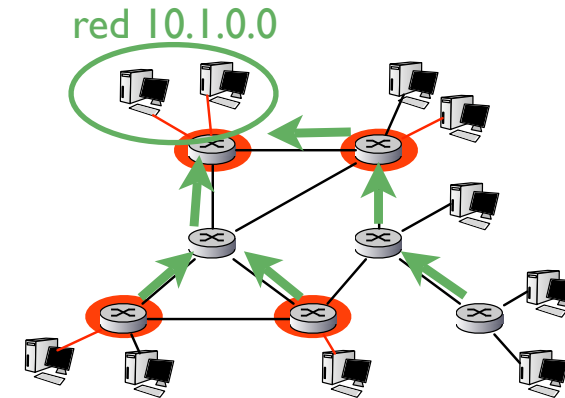
Core Based Trees

DVMRP

- ▶ Distance Vector Multicast Routing Protocol
Parecido a RIP
 - ▶ Mensajes entre routers DVMRP encapsulados en IGMP tipo 0x13
 - ▶ Descubrimiento de vecinos que hablan DVMRP
 - Envío de paquetes Probe a la red de area local grupo All-DVMRP 224.0.0.4
 - ▶ Construcción de tablas de rutas
 - Almaceno las tablas de rutas (red, distancia) para todas las redes
 - Envío periódico (60s) de la tabla de rutas a los vecinos y actualización
 - Metrica 32 = infinito (no se puede llegar a esa red)
No puede manejar redes con mas de 32 saltos entre routers
 - Poisson reverse (metrica+32) en una ruta significa que ese router es mi predecesor en el spanning tree hasta esa ruta
- Construcción automatica de SPTs

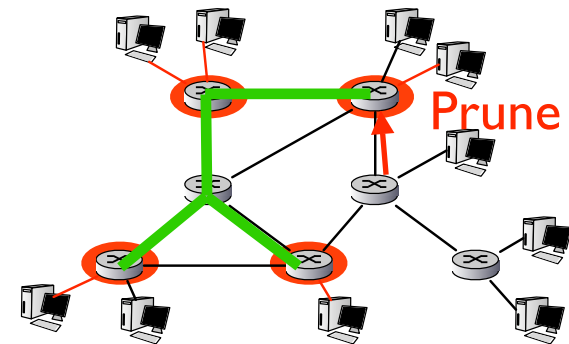
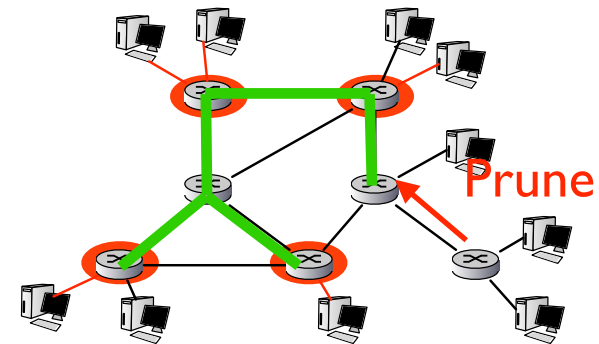
DVMRP

- ▶ Si recibo un paquete multicast de una fuente
 - RPF check: viene por el camino hacia esa fuente?
- ▶ Envío el paquete a los routers que me han hecho poisson reverse para esa fuente
 - = los envío solo hacia abajo por el SPT (fuente, grupo)
- ▶ El trafico multicast inunda el SPT (10.1.1.2, 224.1.1.3)



DVMRP

- ▶ Los routers inundados que
 - no tienen receptores multicast
 - no tienen routers a quien reenviar mandan mensajes DMRP de tipo **Prune** para detener el trafico
- ▶ Si aparecen receptores
 - mensajes **Graft** y **Graft-ack** para reconstruir la rama
- ▶ Los **prunes** caducan en 2 minutos y se vuelve a inundar el arbol
- ▶ Los protocolos de enrutamiento multicast basados en inundacion SPT+pruning se llaman de **modo denso**



DVMRP

- ▶ Enrutamiento multicast de modo denso y basado en distance vector. Es el RIP de los protocolos multicast
- ▶ Ventajas
 - Es simple
 - Soporta CIDR (cosa que RIP no hacia)
- ▶ Desventajas
 - Inestable
 - Metrica 32 no es suficiente para Internet
 - Lleva su propia tabla de rutas
 - No escala muy bien
- ▶ Los routers comerciales implementan DVMRP
- ▶ Mbone usa DVMRP
- ▶ Igual que RIP, es poco aconsejable salvo en entornos reducidos

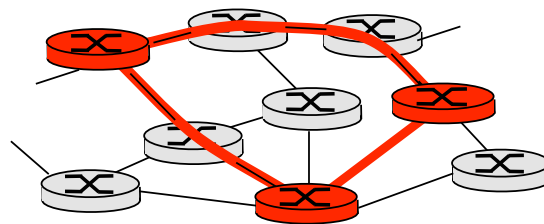
DVMRP

▶ MBone

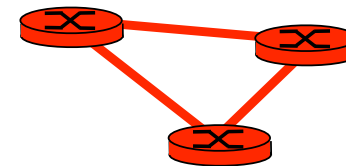
Multicast Backbone

- Red virtual/overlay construida sobre Internet
- Construida para realizar reuniones virtuales
- IETF meeting en SanDiego en Marzo de 1992 uno de los miembros no podia asistir... S. Deering y S. Castner enviaron instrucciones de como hacer tuneles con un ordenador corriendo DVMRP y enviaron a varias instituciones a unirse...

▶ Routers DVMRP conectados entre si por tuneles IP over IP



topología física



topología lógica

PIM

- ▶ Protocol Independent Multicast
 - Enrutamiento multicast independiente del protocolo de enrutamiento
 - No construye su propia tabla de rutas
Usa información de otros protocolos de enrutamiento (OSPF,BGP...)
 - Classless (CIDR) siempre que el protocolo de enrutamiento lo sea
- ▶ Dos modos de funcionamiento
 - **Modo Denso (PIM-DM Dense Mode)**
Inundación+pruning estilo DVMRP
 - **Modo Disperso (PIM-SM Sparse Mode)**
Con shared trees y nodos centrales para formar el arbol

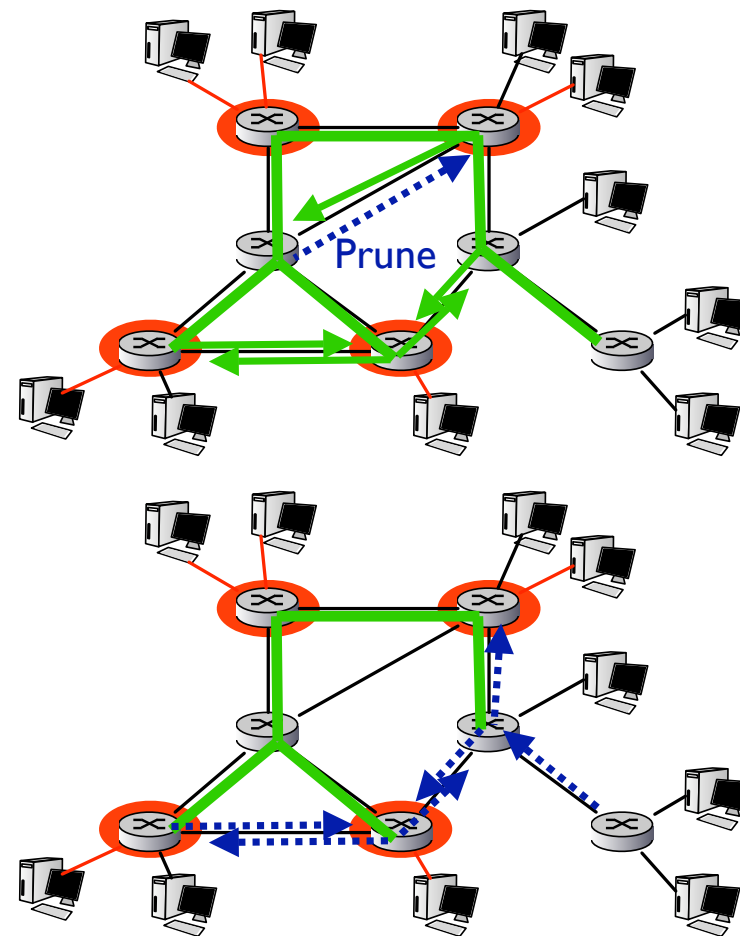
PIM (PIM-DM y PIM-SM)

- ▶ Mensajes PIM
 - Protocolo 103 sobre IP en PIMv2
(en PIMv1 iban dentro de IGMP con tipo 0x13)
- ▶ Descubrimiento de vecinos
 - Mensajes Hello a multicast local 224.0.0.13 All-PIM-Routers cada 30s
 - Las adyacencias se mantienen durante 90s
 - Soporte para elegir un Designated Router (y Querier de IGMPv1)
El de mayor IP salvo que se especifiquen prioridades

PIM-DM

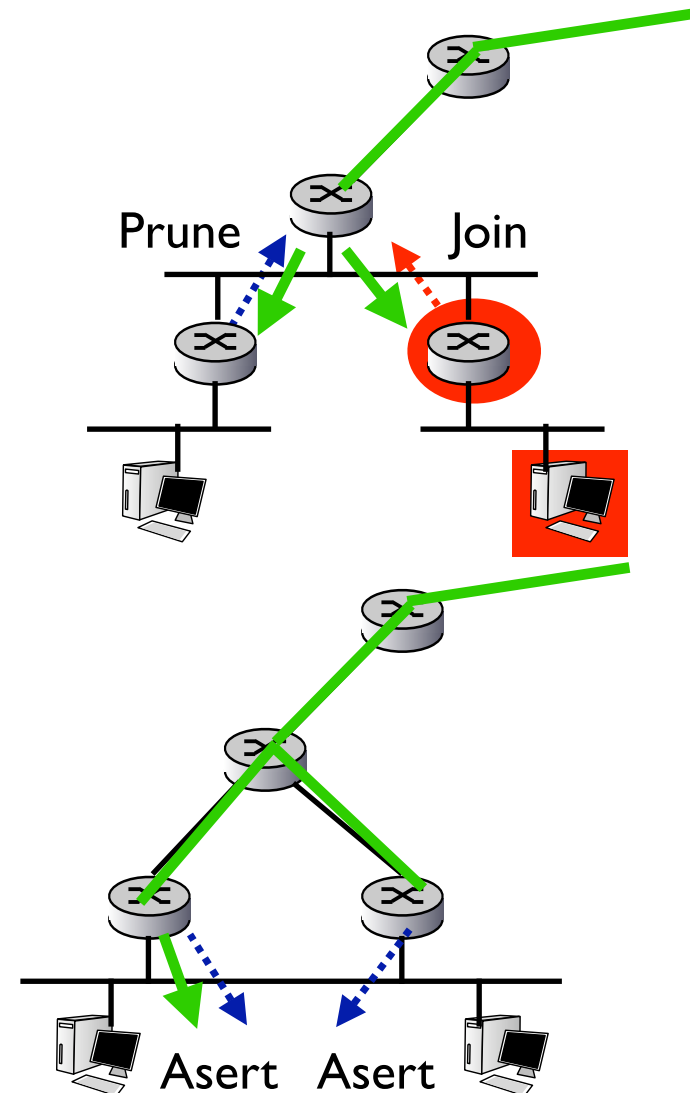
► Modo denso

- Cada router inunda todos los interfaces downstream
- Mensajes PIM Prune cortan los mensajes que no llegan a través del interfaz correcto (RPF)
- Mensajes PIM Prune cortan las ramas del arbol donde no hay receptores
- Tiempo de caducidad del Prune: 2-3 minutos



PIM-DM

- ▶ Prune en LAN
 - Un mensaje de Prune = un vecino quiere hacer Prune
 - Otros vecinos tienen 3s para cancelar el Prune con un mensaje de Join
- ▶ Conflicto de envío en LAN
 - Si veo un paquete multicast en una LAN de un grupo que yo también reenvío
 - Mensajes Assert (con la métrica al origen) gana el que tiene menos distancia
- ▶ Mensajes Graft y Graft-ack
 - Cancelar un prune anterior si aparecen receptores

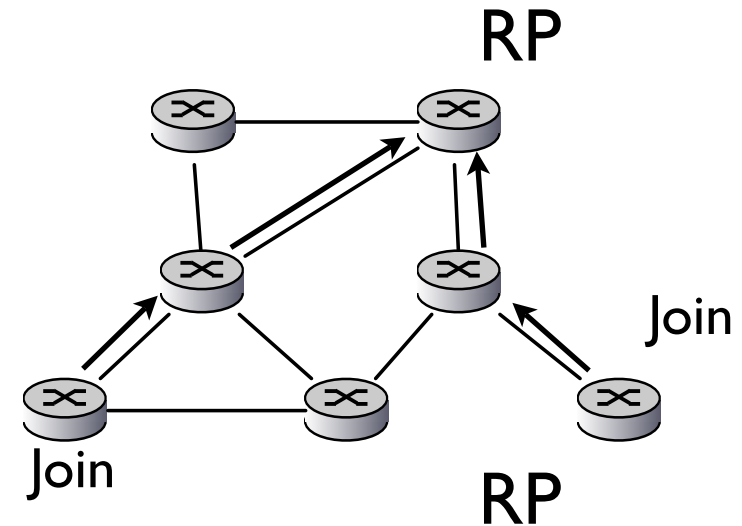


PIM-SM

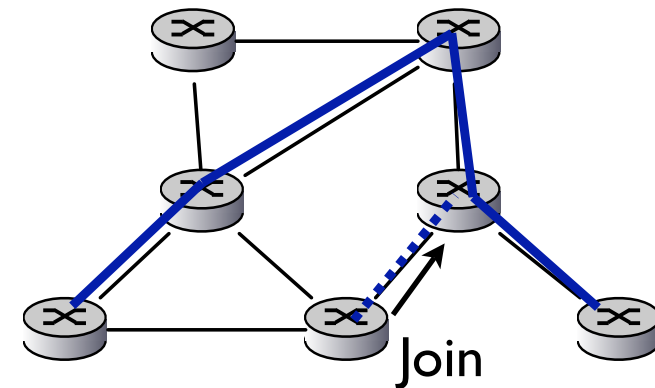
- ▶ Común con PIM-DM
 - Mensajes PIM
 - Descubrimiento de vecinos y Assert
- ▶ Shared trees y router central (RP: Rendezvous point)
- ▶ Estado de los arboles que pasan por los routers en cache multicast
 - Para un arbol (*,grupo) o (fuente,grupo)
 - Interfaz de entrada
 - Interfaces de salida
- ▶ Mensajes PIM Join para construir el arbol al unirse receptores

PIM-SM

- ▶ Unirse al arbol (*,grupo)
 - Envío un mensaje Join al RP
 - Los routers del camino crean la entrada en la cache multicast para (*,grupo) y reenvían el paquete hacia RP
 - Si ya tienen la entrada no hace falta reenviar el paquete

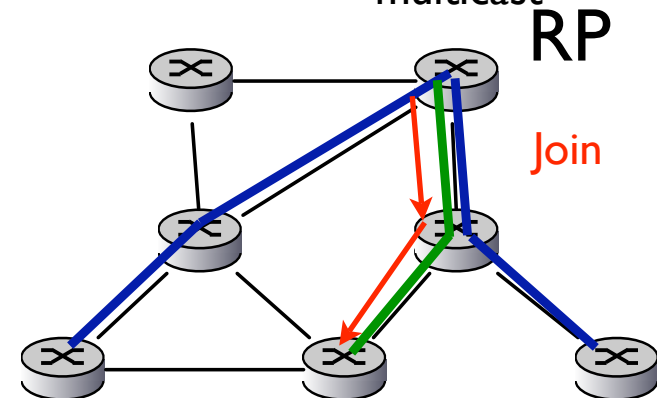
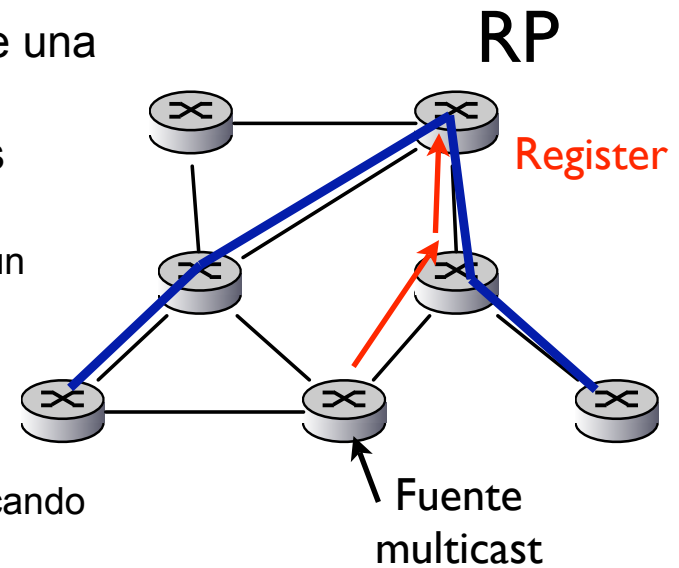


- ▶ Salir del arbol
 - Si no tengo receptores ni otros nodos en direccion downstream
 - Envío un Prune
 - El router upstream elimina la salida y si era la ultima manda otro Prune



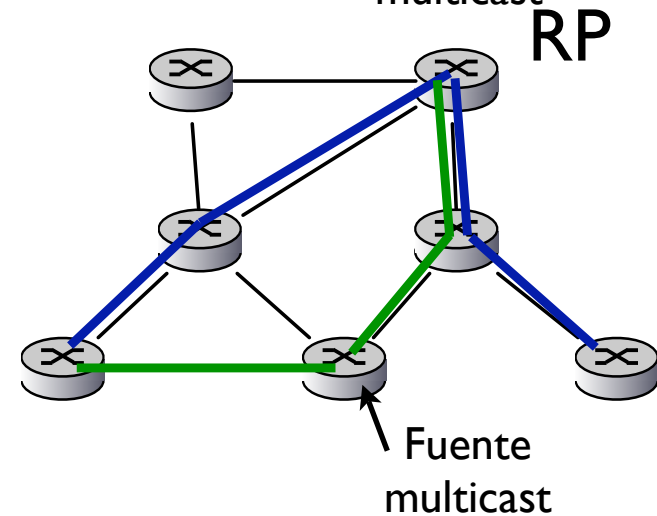
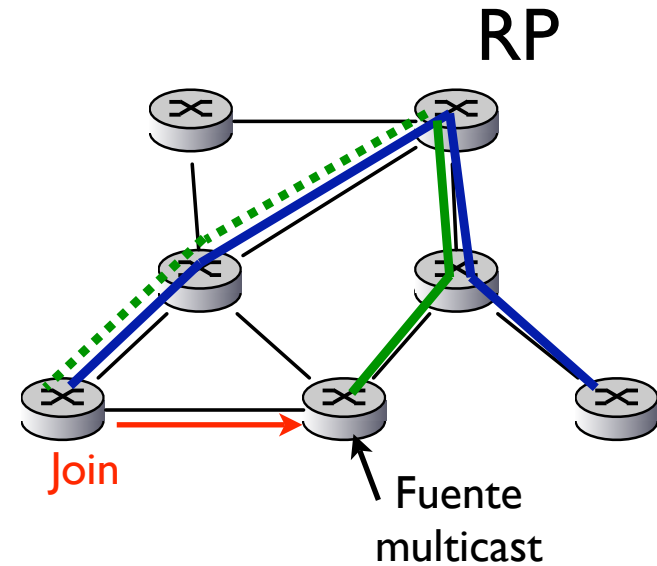
PIM-SM reenvío de multicast

- ▶ Qué recorrido siguen los paquetes multicast desde una fuente?
- ▶ PIM-SM maneja con el mismo método los arboles desde una fuente SPT (fuente, grupo)
 - Los mensajes Join y Prune permiten también unirse a un árbol de fuente
 - El RP se une al árbol SPT de cada fuente que esta transmitiendo en el grupo
- ▶ Cuando una fuente empieza a emitir en multicast
 - El designated router envía mensajes PIM Register indicando la fuente e incluyendo el paquete completo
 - El RP envía los paquetes por el shared tree
 - Envía un Join para unirse al SPT de la fuente
 - Cuando se completa la union se deja de enviar los paquetes en mensajes Register



PIM-SM reenvío de multicast

- ▶ Qué recorrido siguen los paquetes multicast desde una fuente?
- ▶ Los Designated Routers de cada red son responsables de enviar los mensajes Register cuando aparece un emisor multicast
- ▶ Los Designated Routers pueden también unirse al SPT de las fuentes que ven. Se hace por umbral de ancho de banda: si llega más de un determinado tráfico de una fuente me uno al SPT de dicha fuente
- ▶ El umbral suele estar configurado a 0
- ▶ Mensajes Prune especiales para permitir hacer prune del shared tree referido a una sola fuente



PIM resumen

- ▶ Dos modos de operación
- ▶ Modo denso (PIM-DM)
 - Muchos miembros del grupo próximos
 - Ancho de banda abundante
 - Por defecto se considera a los routers miembros del grupo hasta que hagan pruning
 - Arbol multicast construido desde la fuente (SPT)
- ▶ Modo disperso (PIM-SM)
 - Miembros del grupo dispersos
 - Ancho de banda no tan abundante
 - Por defecto se considera a los routers no miembros, hasta que se unen
 - Árboles multicast construido desde el receptor (con nodo central) y SPTs
 - Queda el problema de configurar/encontrar los RPs

Conclusiones

- ▶ Conceptos básicos de broadcast y multicast
 - Basados en arboles compartidos
 - Basados en arboles por fuente
- ▶ Multicast en IP
 - Direcciones de grupos multicast
 - Gestión de miembros del grupo: IGMP
- ▶ Enrutamiento multicast en Internet
 - DVMRP y Mbone
 - PIM-DM
 - PIM-SM