

# Capítulo 9 Multimedia en Redes de Computadores

Material basado en el Texto:  
7° Edition Computer Networking: A Top Down  
Approach,  
Jim Kurose, Keith Ross.

# Multimedia en Redes de Computadores

9.1 Aplicaciones Multimedia en Redes

9.2 Streaming de Video almacenado

9.3 voice-over-IP

9.4 Protocolos de aplicaciones de conversaciones en *tiempo-real*

9.5 Soorte de red para multimedia

# Voice-over-IP (VoIP)

- *Requerimientos extremo-a-extremo en VoIP*: aspectos necesarios para mantener una conversación:
  - Altos retardos son notorios, dificulta interactividad
  - < 150 msec: bueno
  - > 400 msec malo, debe evitars asumiendo pérdida
  - Retardo debe incluir retardos de la aplicación (paquetizar, reproducción) y de la red
- *Iniciación de la sesión*: cómo se da aviso de la IP, puerto, algoritmo de codificación?
- *Servicios de valor agregado*: reenvío de llamados, grabación, identificación del llamador
- *Servicios de emergencia*: 911 (USA) 131, 132, 133 (Chile)

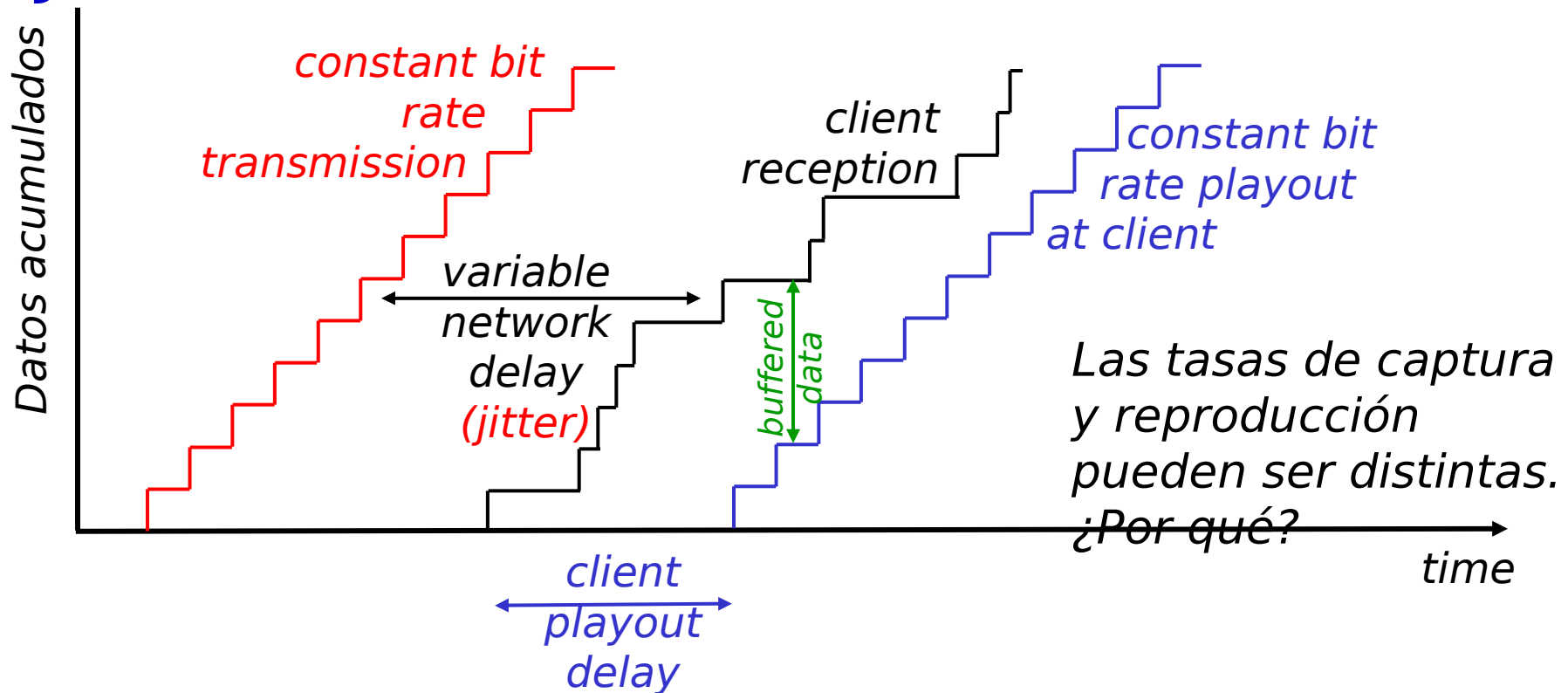
# Características de VoIP

- ❑ Audio emisor: alterna habla con periodos de silencio.
  - 64 kbps durante momentos de habla
  - Paquetes son generados sólo durante el habla
  - Segmento de 20 msec a 8 Kbytes/sec: 160 bytes de datos
- ❑ Encabezado capa aplicación es agregado a cada segmento (Protocolo RTP).
- ❑ Segmento + encabezado es encapsulado en datagrama UDP o segmento TCP.
- ❑ Aplicación envía datagrama UDP por el socket cada 20 ms durante habla.

# VoIP: Pérdidas y retardo

- ❑ **Pérdidas en la red:** pérdida de datagrama IP debido a congestión en la red (overflow de buffer de router)
- ❑ **Pérdida por retardo:** Datagrama IP llega muy tarde para su reproducción en el receptor
  - retardo: procesamiento, colas en red; retardo en sistemas extremos (Tx y Rx)
  - Retardo máximo tolerable típico: 400 ms
- ❑ **Tolerancia a pérdidas:** dependiendo de codificación de voz, se puede tolerar entre 1% y 10% de paquetes perdidos.

# Variaciones del retardo (Delay Jitter)



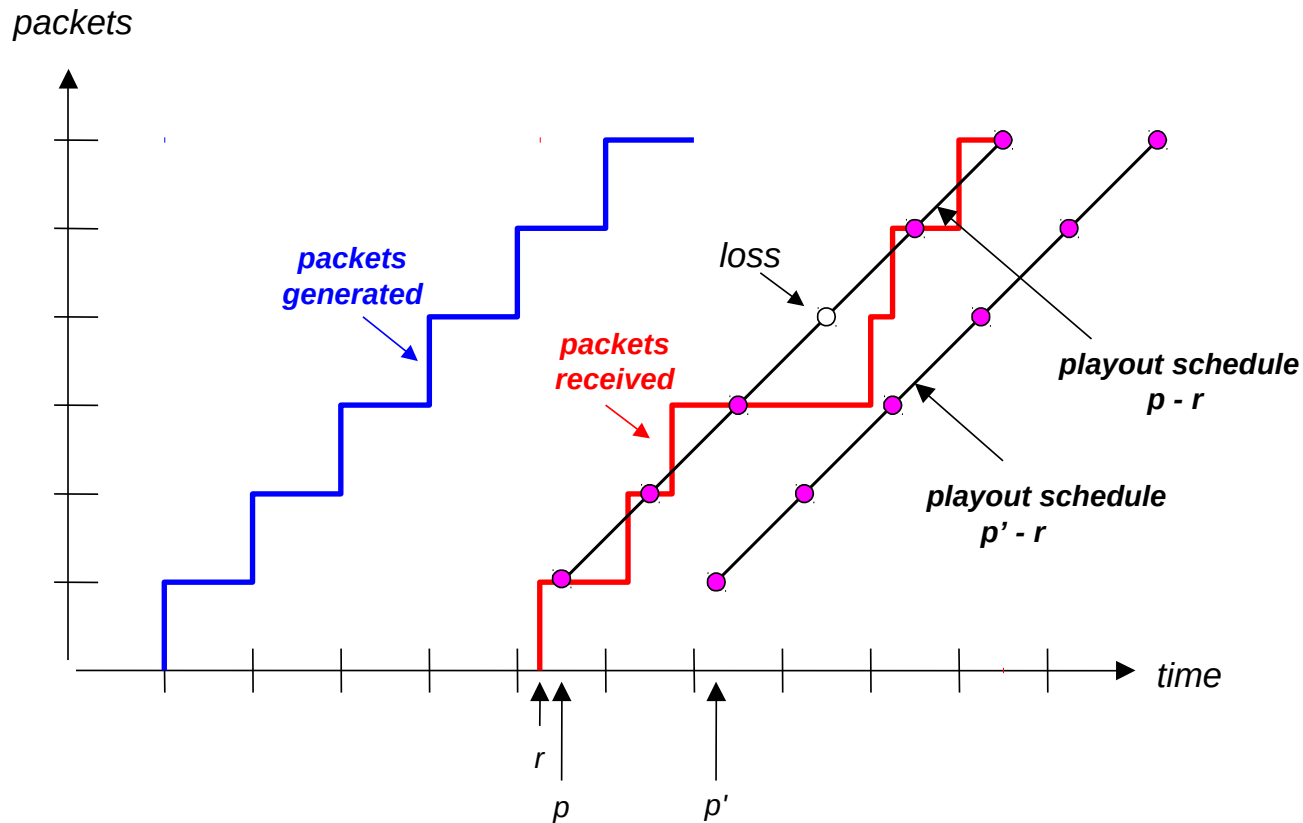
- Consideremos retardo extremo a extremo de dos paquetes consecutivos: **diferencia** puede ser más o menos de 20 ms

# VoIP: Retardo de reproducción fijo

- ❑ Receptor intenta reproducir cada golpe de habla exactamente  $q$  [ms] después que el habla fue generada.
  - habla tiene marca de tiempo  $t$ : reproducir después de  $t+q$  .
  - Habla llega después de  $t+q$ : datos llegan muy tarde para reproducción, datos son “perdidos”
- ❑ Compromiso para  $q$ :
  - $q$  grande: menor pérdida de paquete, más retardo
  - $q$  pequeño: mejor experiencia interactiva

# Retardo de reproducción fijo

- *Tx genera paquetes cada 20 ms durante habla.*
- *Primer paquete recibido en tiempo  $r$*
- *Primer itinerario de reproducción: comienza en  $p$*
- *Segundo itinerario de reproducción: comienza en  $p'$*





# Retardo de reproducción Adaptativo, I

- **Objetivo:** minimizar retardo de reproducción, manteniendo baja la tasa de pérdida por retardo
- **Estrategia:** Ajuste del retardo de reproducción adaptativo:
  - Retardo de red estimado, ajustar el retardo de reproducción al comienzo de cada segmento de habla.
  - Periodos de silencio alargados o comprimidos.
  - Habla aún reproducida cada 20 ms durante su presencia.

$t_i$  = marca de tiempo de  $i^{\text{mo}}$  paquete

$r_i$  = tiempo recepción paquete  $i$

$p_i$  = tiempo paquete  $i$  es reproducido (*playout*)

$r_i - t_i$  = retardo de red para  $i^{\text{mo}}$  paquete

$d_i$  = retardo promedio estimado después de recibir  $i^{\text{mo}}$  paquete

*Estimación dinámica de retardo promedio en receptor, ojo con valores que adopta:*

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

*Donde  $u$  es una constante fija (e.g.,  $u = .01$ ).*

# Retardo de Reproducción Adaptativo II

- También es útil estimar el promedio de las variaciones de retardo,  $v_i$ :

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

- Los estimadores  $d_i$  y  $v_i$  son calculados para cada paquete recibido, aún cuando ellos son usados sólo al inicio de cada segmento de habla.

- El primer paquete de un segmento de habla es reproducido en tiempo:

$$p_i = t_i + d_i + Kv_i$$

Donde  $K$  es una constante positiva (ej. 4).

- Si la reproducción es bajo demanda o en vivo no interactiva, podemos usar mayor  **$K$**
- Paquetes restantes dentro de un segmento de habla son reproducidos periódicamente (por ejemplo, cada 20 [ms]).

# Reproducción adaptativo, III

**Q:** Cómo el receptor determina que un paquete es el primero en un segmento de habla?

- ❑ Si no hay pérdida, receptor mira marcas de tiempo sucesivas.
  - Diferencia de marcas de tiempo sucesivas  $> 20$  ms  $\Rightarrow$  comienza nuevo segmento de habla.
- ❑ Con posible pérdida, receptor debe mirar las marcas de tiempo y números de secuencia.
  - Diferencia de marcas de tiempo sucesivas  $> 20$  ms **y** números de secuencia sucesivos espacios  $\Rightarrow$  segmento de habla comienza. Ésta es la regla general.
- ❑ Se requiere detección del habla en transmisor. Más adelante.

# Recuperación de pérdidas de paquetes (1)

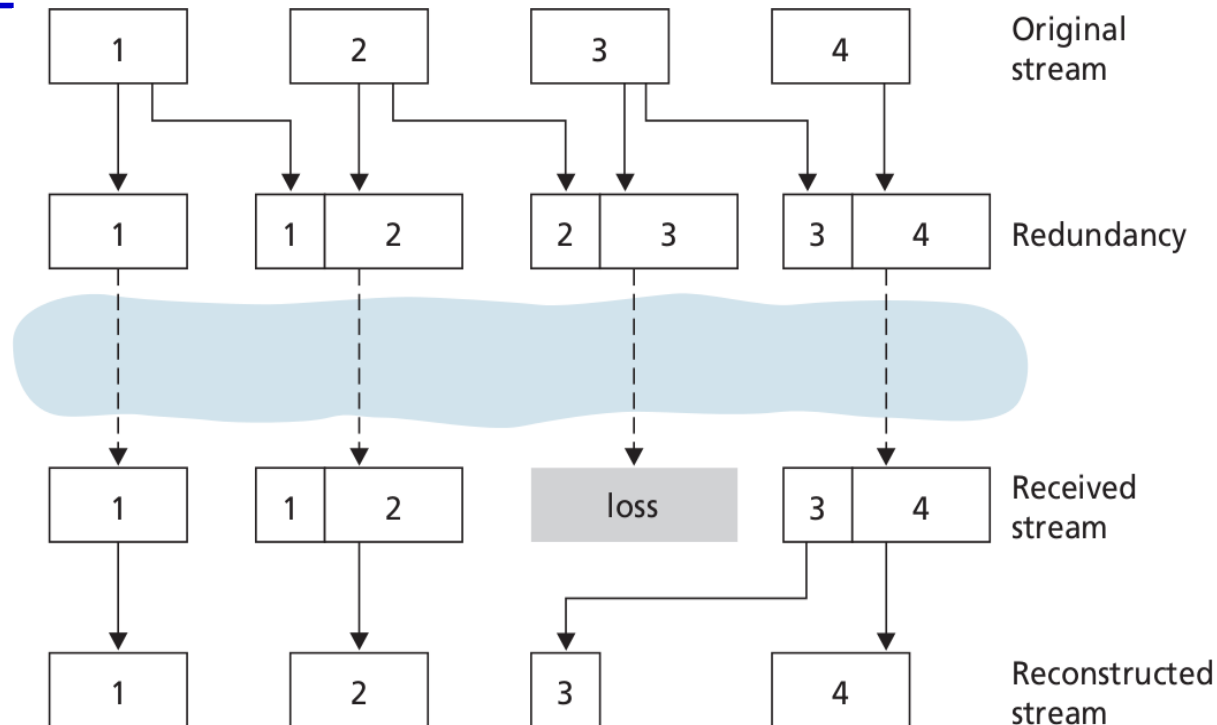
**Desafío:** Recuperarse de pérdidas de paquetes dada el pequeña retardo tolerable entre transmisión en fuente y reproducción.

- ❑ Cada ACK/NAK toma  $\sim 1$  RTT
- ❑ Alternativa: forward error correction (FEC)
  - Enviar suficientes bits para permitir la recuperación sin retransmisión.
- ❑ Esquema FEC simple
- ❑ Por cada  $n$  paquetes crea un paquete redundante dando paridad a los  $n$  originales.
- ❑ envía  $n+1$  paquetes, aumenta ancho de banda ocupado en factor  $1/n$ .
- ❑ Se puede reconstruir los  $n$  paquetes originales si hay a lo más un paquete perdido de los  $n+1$

# Recuperación de paquetes perdidos (2)

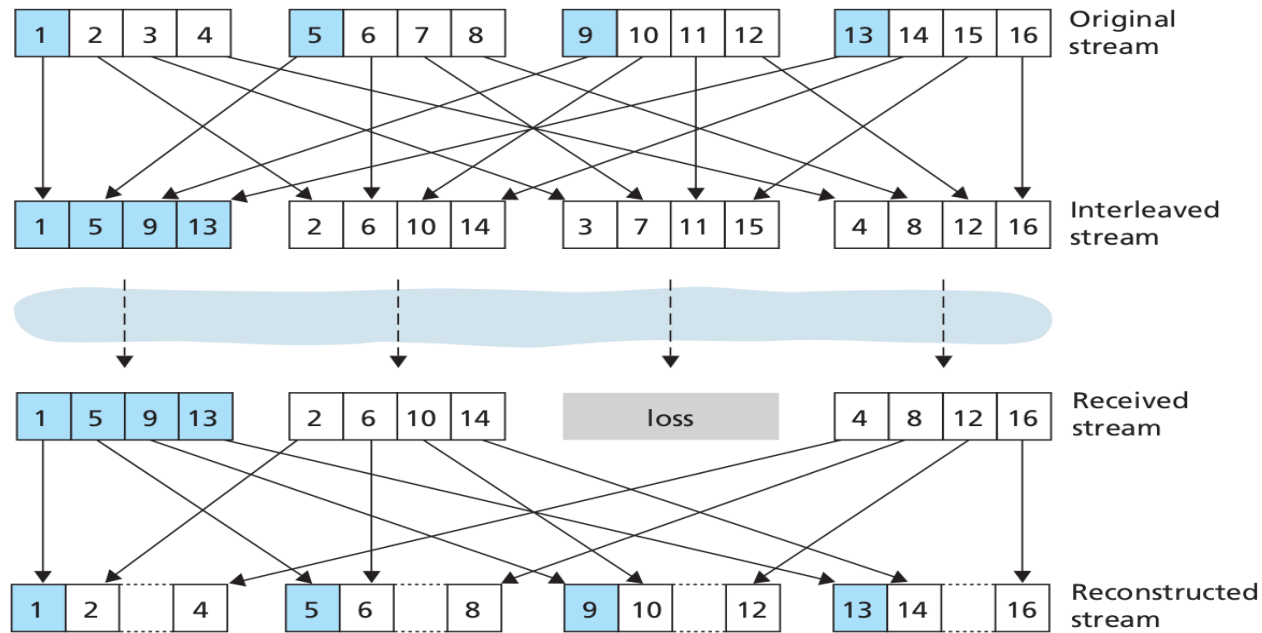
## 2º esquema FEC

- agrega un flujo de baja calidad
- envía flujo de baja resolución como información redundante
- por ejemplo, flujo nominal PCM a 64 kbps y flujo redundante GSM a 13 kbps.



- Cuando no hay pérdidas consecutivas, el receptor puede subsanar la pérdida.
- Se puede agregar también las tramas de baja calidad (n-1) y (n-2)

# Recuperación de pérdida de paquetes (3)



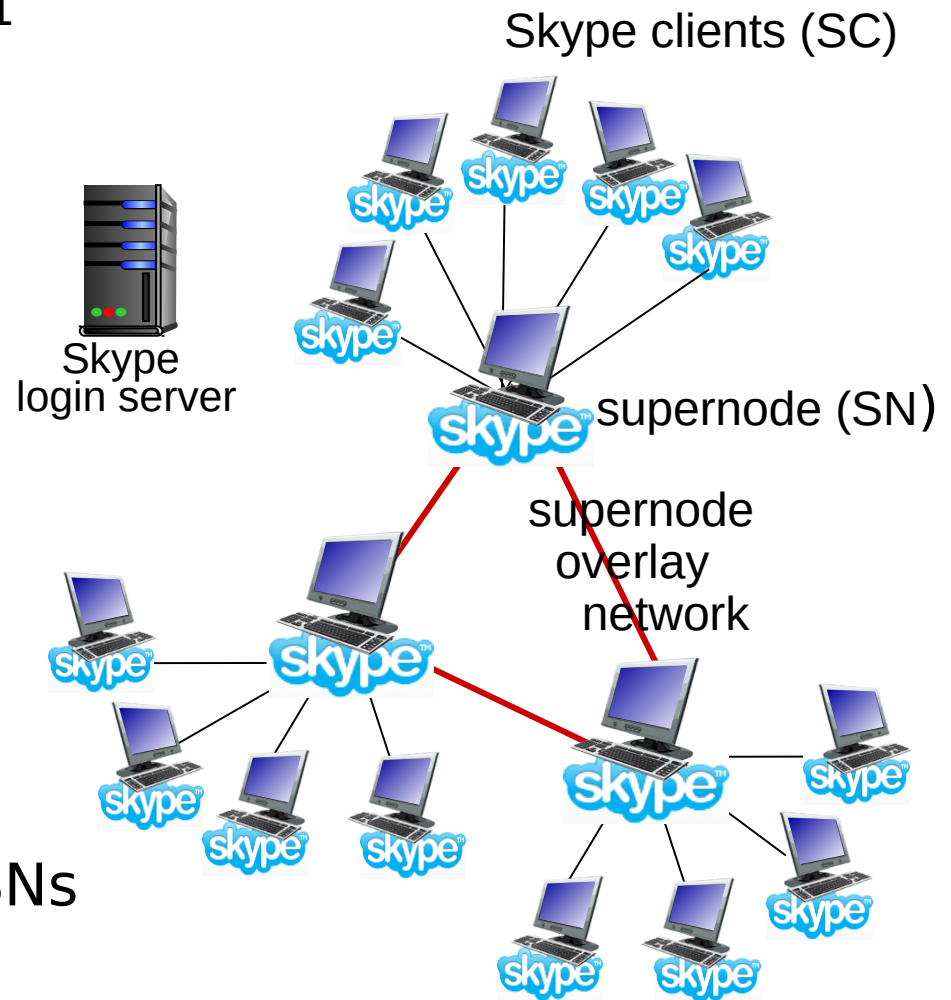
## Entrelazado

- ❑ Tramas son subdivididas en pequeñas unidades
- ❑ Por ejemplo, unidades de 4 ó 5 ms
- ❑ Paquete contiene pequeñas unidades de tramas diferentes
- ❑ Si paquete se pierde, aún se tiene la mayoría de cada trama
- ❑ No hay redundancia
- ❑ Se agrega retardo de reproducción

# Caso estudio: Voice-over-IP:

## Skype

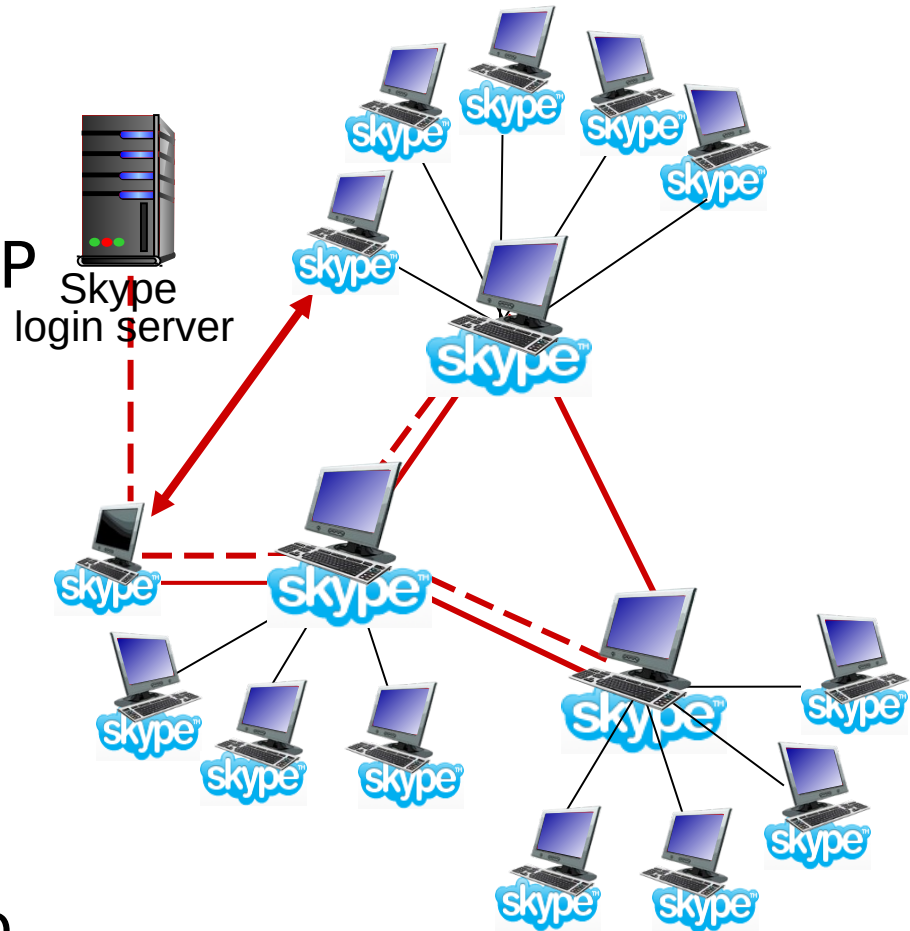
- Comprado por Microsoft el 2011 por sobre US\$8 billones
- Protocolo de aplicación propietario (inferido vía ingeniería inversa)
  - Mensajes encriptados
- Componentes P2P:
  - **Cientes (SC):** son los pares skype comunicados vía llamado VoIP
  - **super nodes (SN):** pares skype con funciones especiales
  - **overlay network:** entre SNs para localizar SCs
  - **login server**



# P2P voice-over-IP: skype

## Operación de clientes skype:

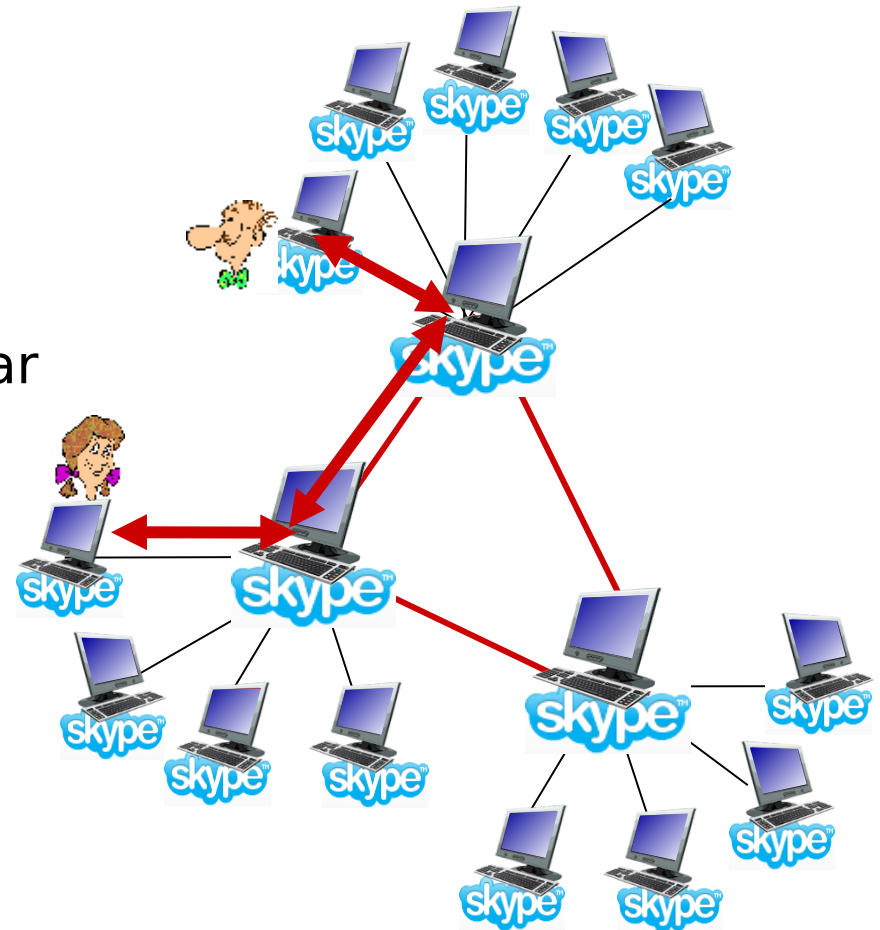
1. Ingresa a red contactando un SN (IP address cached) usa TCP
2. logs-in (username, password) a servidor central skype
3. obtiene IP de llamado desde SN, SN overlay
4. Inicia llamada directamente al llamado





# Skype: pares y relays

- **problema:** Alice y Bob están detrás de “NATs”
  - NAT previene inicio de conexiones a pares internos
  - Par interno sí puede iniciar conexión hacia afuera.
- ❖ **Solución relay:** Alice y Bob mantienen conexiones abiertas a sus SNs
  - Alice pide a su SN conectarse a Bob
  - SN de Alice se conecta a SN de Bob
  - SN de Bob se conecta a Bob por su conexión previamente abierta.



# Multimedia en Redes de Computadores

9.1 Aplicaciones Multimedia en Redes

9.2 Streaming de Video almacenado

9.3 voice-over-IP

9.4 Protocolos de aplicaciones de conversaciones en *tiempo-real*

9.5 Soorte de red para multimedia