QoS: Introducción

Área de Ingeniería Telemática
http://www.tlm.unavarra.es

Máster en Comunicaciones

Objetivos

- Comprender a qué llamaremos “calidad”
- Recordar los parámetros de red que vamos a medir y valorar

¿ Qué es esto de la calidad ?

Para el usuario final

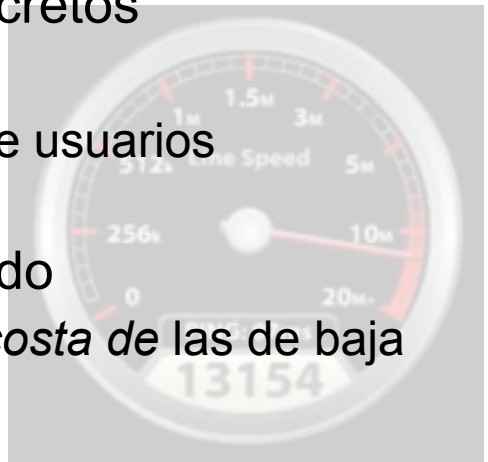
- Para un usuario experimentado es normal que una llamada con un ordenador tenga diferente calidad que una por teléfono fijo o que una por móvil
- ¡ Aunque todas se cursen por la misma red !
- Es simplemente aquello a lo que está acostumbrado
- Si nunca ha usado un móvil esperará una calidad similar a la PSTN y se quejará
- Lo mismo si nunca ha usado VoIP
- La calidad es relativa a las expectativas
- Lo mismo con el precio, si está acostumbrado a una tarifa plana o gratis le extrañará pagar

(...)

¿ Qué es esto de la calidad ?

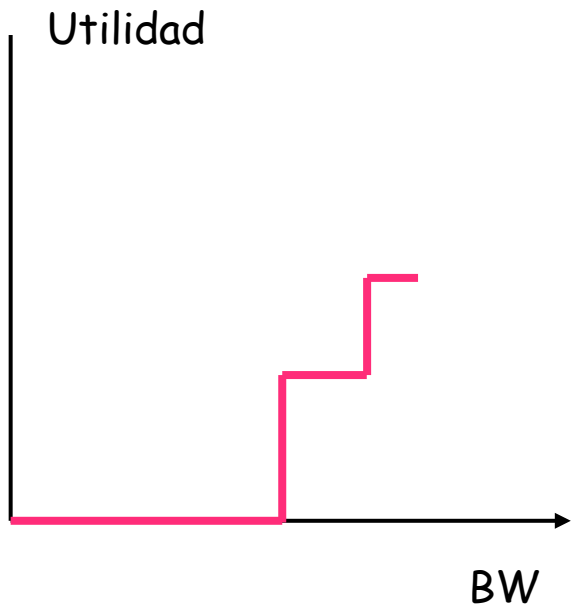
Para el técnico

- Habilidad de la red de *diferenciar* a unos determinados tipos de tráfico, probablemente de unos servicios concretos
- Controlar ciertos parámetros estadísticos:
  - Bandwidth, pérdidas, retardo, jitter… quejas de usuarios
  - Más absolutos y medibles
- Se basa en un reparto “injusto” pero controlado
  - Ofrecer recursos a clases de alta prioridad *a costa de* las de baja
- Formalizados en SLAs
  - Dentro varios SLSs (*Service Level Specifications*)
  - Acuerdo entre proveedor de servicio (la red) y el suscriptor (el cliente)
  - Especifica la calidad de servicio que garantizará el proveedor
  - La red mantendrá su promesa mientras los flujos de usuario se mantengan dentro de su especificación de tráfico
  - Especifica las medidas que se tomarán si se incumple
  - Gran cantidad de parámetros posibles según el servicio



Usuario: Utilidad

- Aplicaciones son sensibles a pérdidas, capacidad, retardo, variación en el retardo
- Por debajo de un umbral puede no ser útil el tráfico
- Ofrecer garantías de prestaciones para
  - Que el usuario esté satisfecho
  - Que los recursos se usen de forma óptima



¿ Quién necesita QoS ?

- Dos tipos de aplicaciones/tráfico:
  - Elástico
    - Se ajusta ante grandes cambios en retardo y throughput
    - Sigue manteniendo la funcionalidad de la aplicación
  - Inelástico
    - Si no se cumplen unos requisitos de calidad la utilidad se vuelve 0

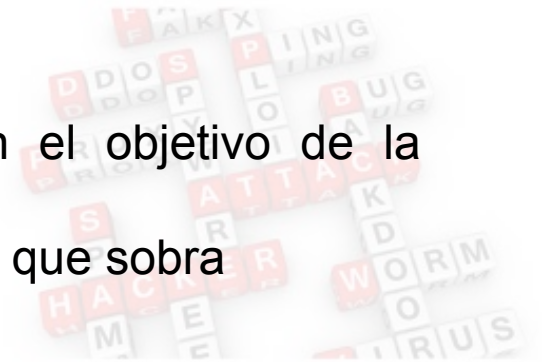
Requisitos de QoS de las aplicaciones

Aplicación	Fiabilidad	Retardo	Jitter	Ancho de Banda
Correo electrónico	Alta (*)	Alto	Alto	Bajo
Transferencia de ficheros	Alta (*)	Alto	Alto	Medio (**)
Acceso Web	Alta (*)	Medio	Alto	Medio
Login remoto	Alta (*)	Medio	Medio	Bajo
Audio bajo demanda	Media	Alto	Medio	Medio
Vídeo bajo demanda	Media	Alto	Medio	Alto
Telefonía	Media	Bajo	Bajo	Bajo
Vídeoconferencia	Media	Bajo	Bajo	Alto

- (*) La fiabilidad alta en estas aplicaciones se consigue automáticamente al utilizar el protocolo de transporte TCP
- (**) Transferencia de ficheros: si es interactiva el usuario espera que tarde proporcionalmente al tamaño, luego depende del BW

¿ Quién necesita QoS ?

- Voz (IP telephony, radio?)
- Vídeo (streaming, videoconferencia)
- Ciertas aplicaciones de datos (generalmente elásticas)
  - *Transactional Data/Interactive Data* (SAP, Oracle…)
  - *Bulk Data* (backups, replicación en redes de contenidos…)
  - *Locally Defined Mission-Critical Data* (mayor que *transactional*)
- Resto:
  - *Best Effort*
  - Dejar BW para él
  - Gran cantidad de aplicaciones en una empresa (centenares)
  - Probablemente no se puedan clasificar todas, ¡no ahogarlas!
- ¿Queda algo?: *Scavenger Service*
  - *Less than BE*
  - Tráfico no deseado: DoS, Worms, etc
  - Web surfing a destinos no relacionados con el objetivo de la empresa
  - Si no se descarta se cursa solo en la capacidad que sobra

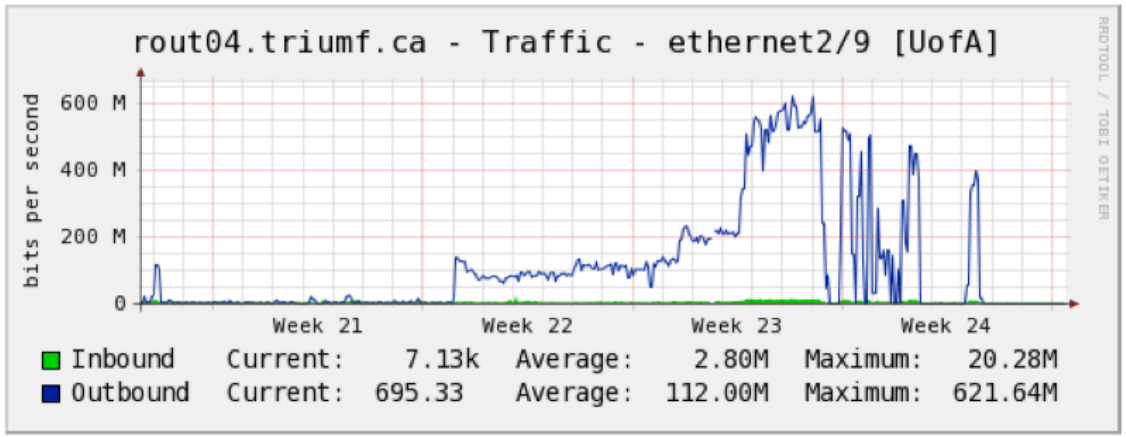


¿ Qué necesita ?

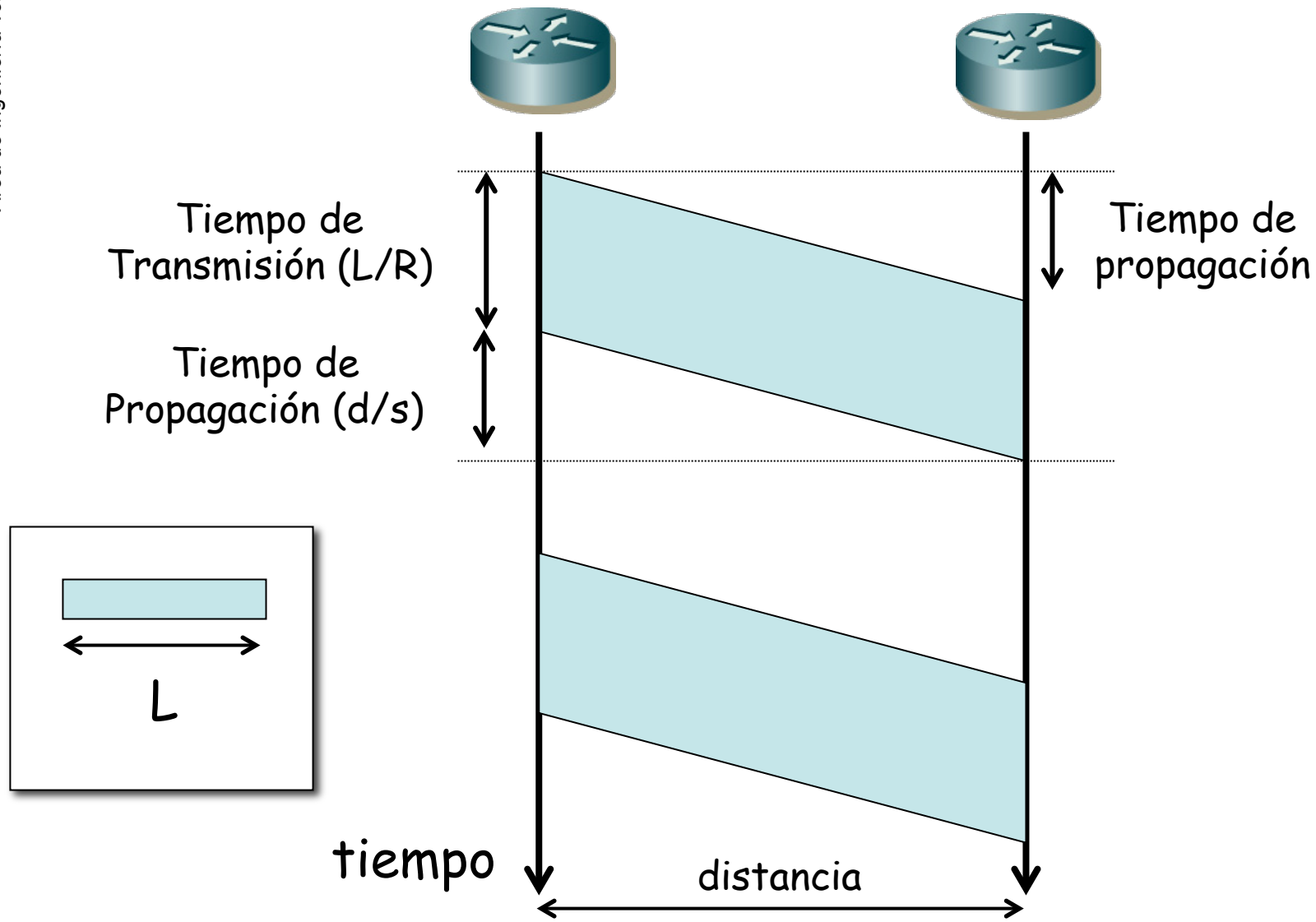
- Que sea predecible el comportamiento de la red
- Garantizar (depende de la aplicación):
  - Bandwidth (Throughput)
  - Delay
  - Variación en el retarto (jitter)
  - Pérdidas

Bandwidth

- La cantidad reservada del BW del canal
- También llamado *Throughput*
  - Throughput instantáneo: tasa a la cual se transmiten o transfieren o reciben datos
  - Throughput medio: cantidad de datos transferidos en un intervalo de tiempo divididos por ese tiempo
  - Ejemplo: transferencia de fichero de tamaño F bits en un tiempo T segundos ha sido a F/T bps
  - En realidad, throughput instantáneo medido por debajo del tiempo de un paquete es el bit rate del enlace
  - Medido por encima de esa escala es un throughput medio

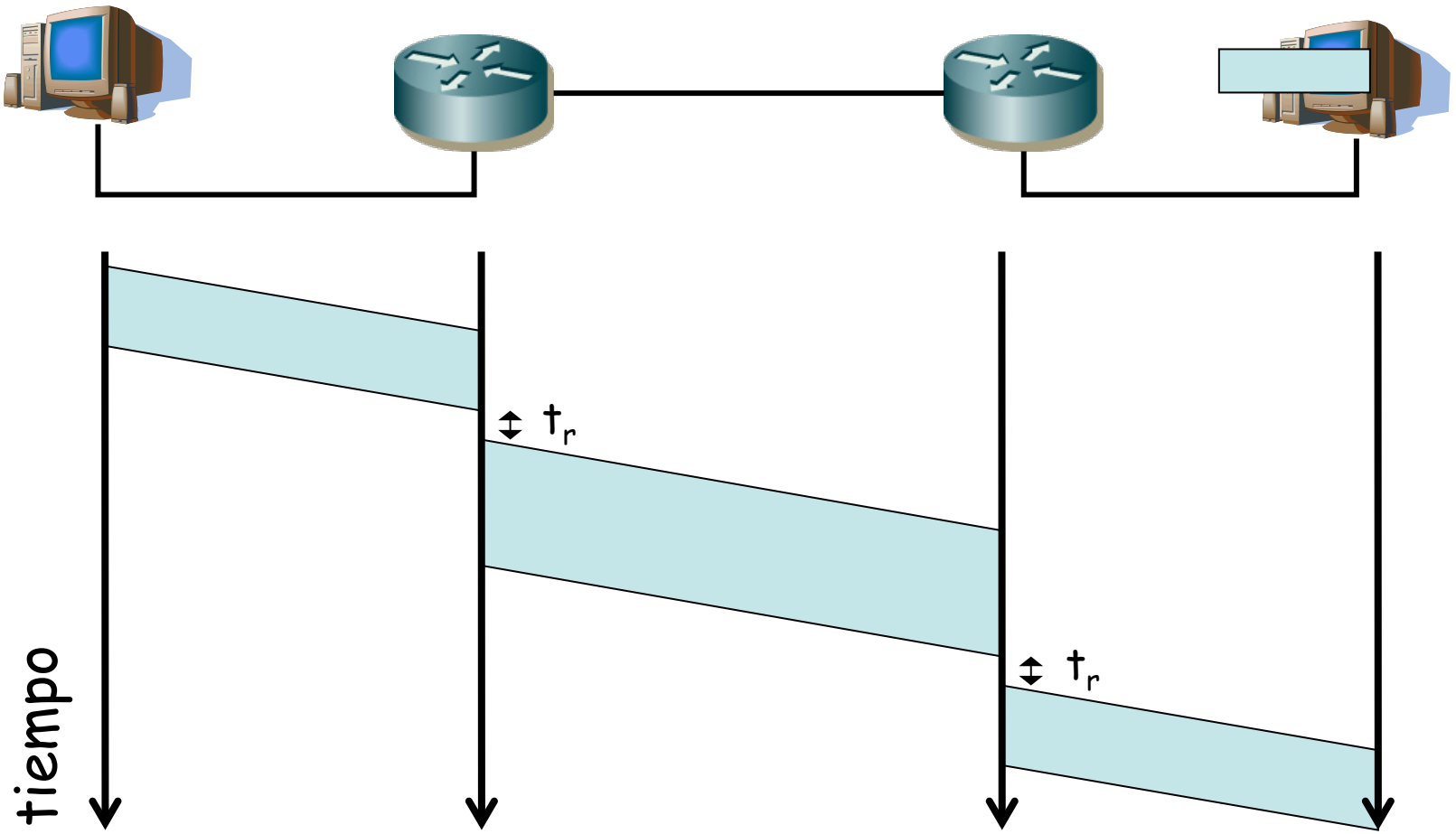


Retardos de transmisión y propagación



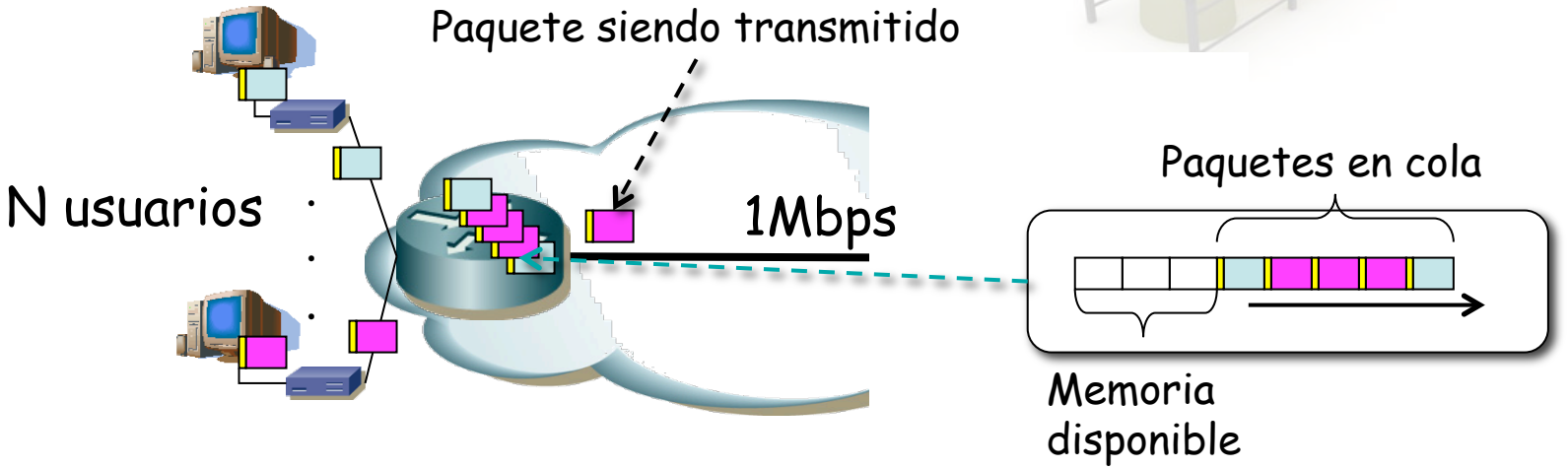
Tiempo de procesado

- El conmutador debe tomar una decisión para cada paquete, la cual lleva tiempo (t_r)



Retardo en cola

- Los paquetes pueden llegar al router a una velocidad mayor que la capacidad del enlace de salida
- O pueden llegar varios simultáneamente por enlaces diferentes pero solo puede salir uno a la vez
- El router los almacena en memoria hasta poder enviarlos
- Esperan en una *cola* (normalmente en el interaz de salida)
- Si no queda espacio en memoria para almacenar un paquete, normalmente éste se pierde (*drop-tail policy*)



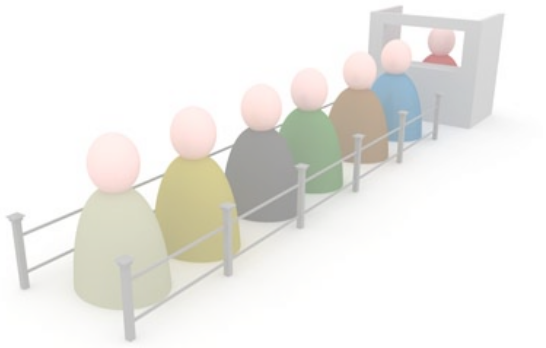
Retardo en cola

- R = tasa de transmisión
- L = longitud del paquete
- λ = tasa media de llegadas por segundo

- Llegan λ paquetes por segundo
- Llegan λL bps

Intensidad del tráfico:

$$I = \frac{\lambda L}{R}$$



Si I > 1

- Llega más tráfico del que se puede cursar
- La cola crece indefinidamente
- Pérdidas al llenarse la cola del interfaz de salida

Retardo en cola

- R = tasa de transmisión
- L = longitud del paquete
- λ = tasa media de llegadas por segundo

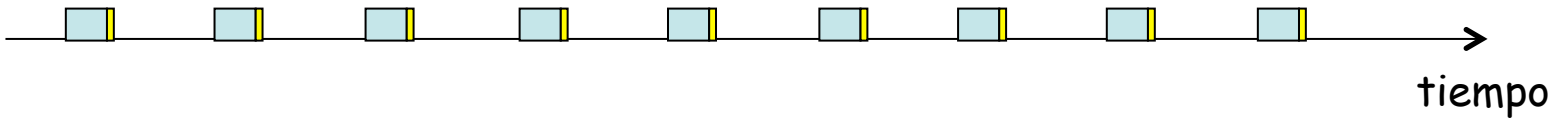
- Llegan λ paquetes por segundo
- Llegan λL bps

Intensidad del tráfico:

$$I = \frac{\lambda L}{R}$$

Si I < 1 y llegadas periódicas

- Supongamos paquetes de igual tamaño
- El tiempo de transmisión es menor al tiempo entre llegadas
- No se forma cola



Retardo en cola

- R = tasa de transmisión
- L = longitud del paquete
- λ = tasa media de llegadas por segundo

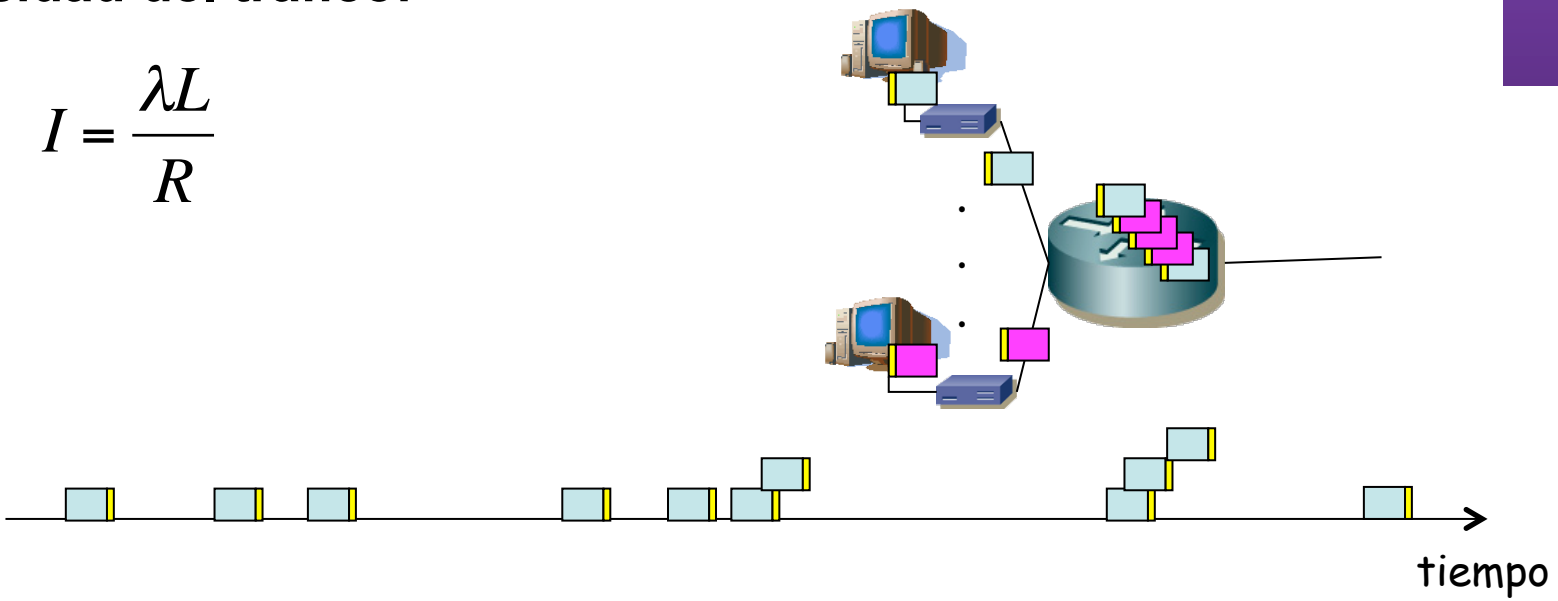
- Llegan λ paquetes por segundo
- Llegan λL bps

Intensidad del tráfico:

$$I = \frac{\lambda L}{R}$$

Si I < 1 y llegadas "aleatorias"

- En media entra menos tráfico del que puede salir
- Pero pueden llegar dos paquetes muy próximos
- Se forma cola
- Depende de cómo lleguen los paquetes y sus tamaños (...)



Retardo en cola

- R = tasa de transmisión
- L = longitud del paquete
- λ = tasa media de llegadas por segundo

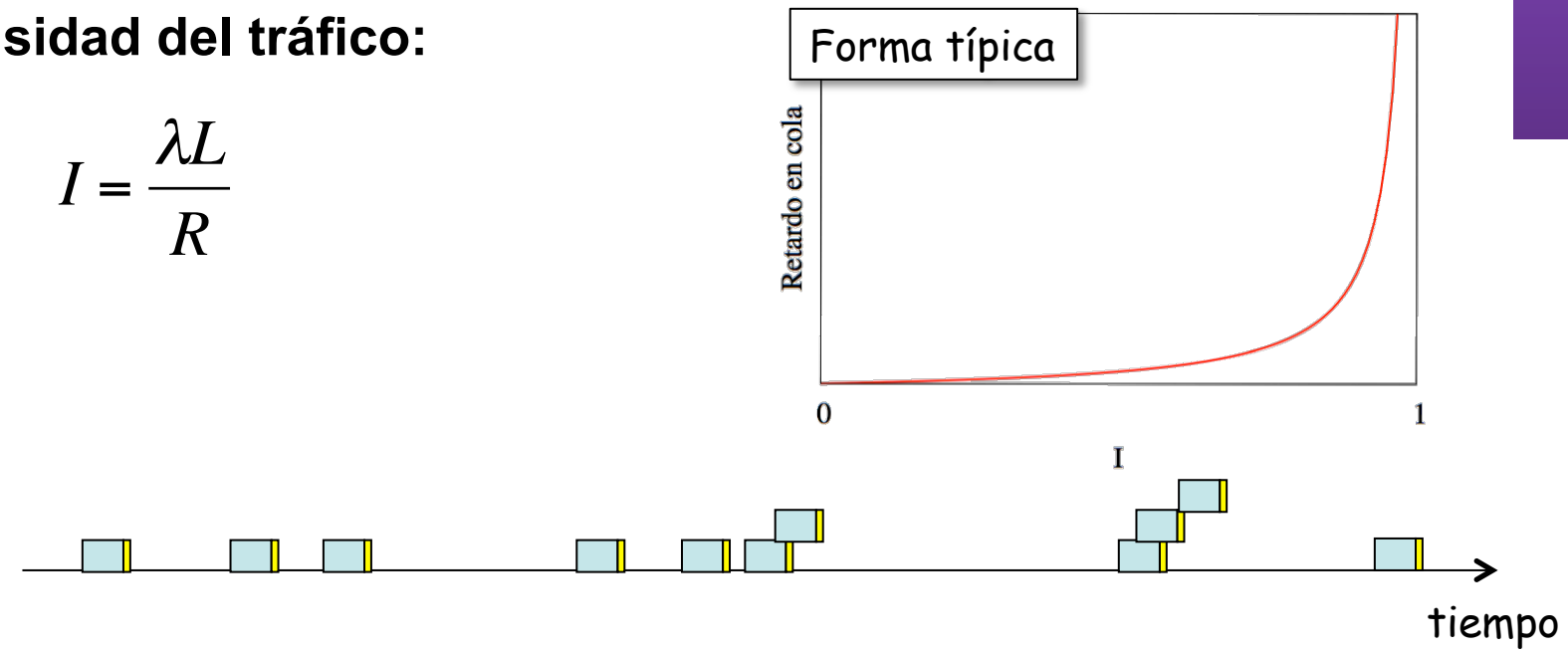
- Llegan λ paquetes por segundo
- Llegan λL bps

Intensidad del tráfico:

$$I = \frac{\lambda L}{R}$$

Si I < 1 y llegadas "aleatorias"

- En media entra menos tráfico del que puede salir
- Pero pueden llegar dos paquetes muy próximos
- Se forma cola
- Depende de cómo lleguen los paquetes y sus tamaños (...)

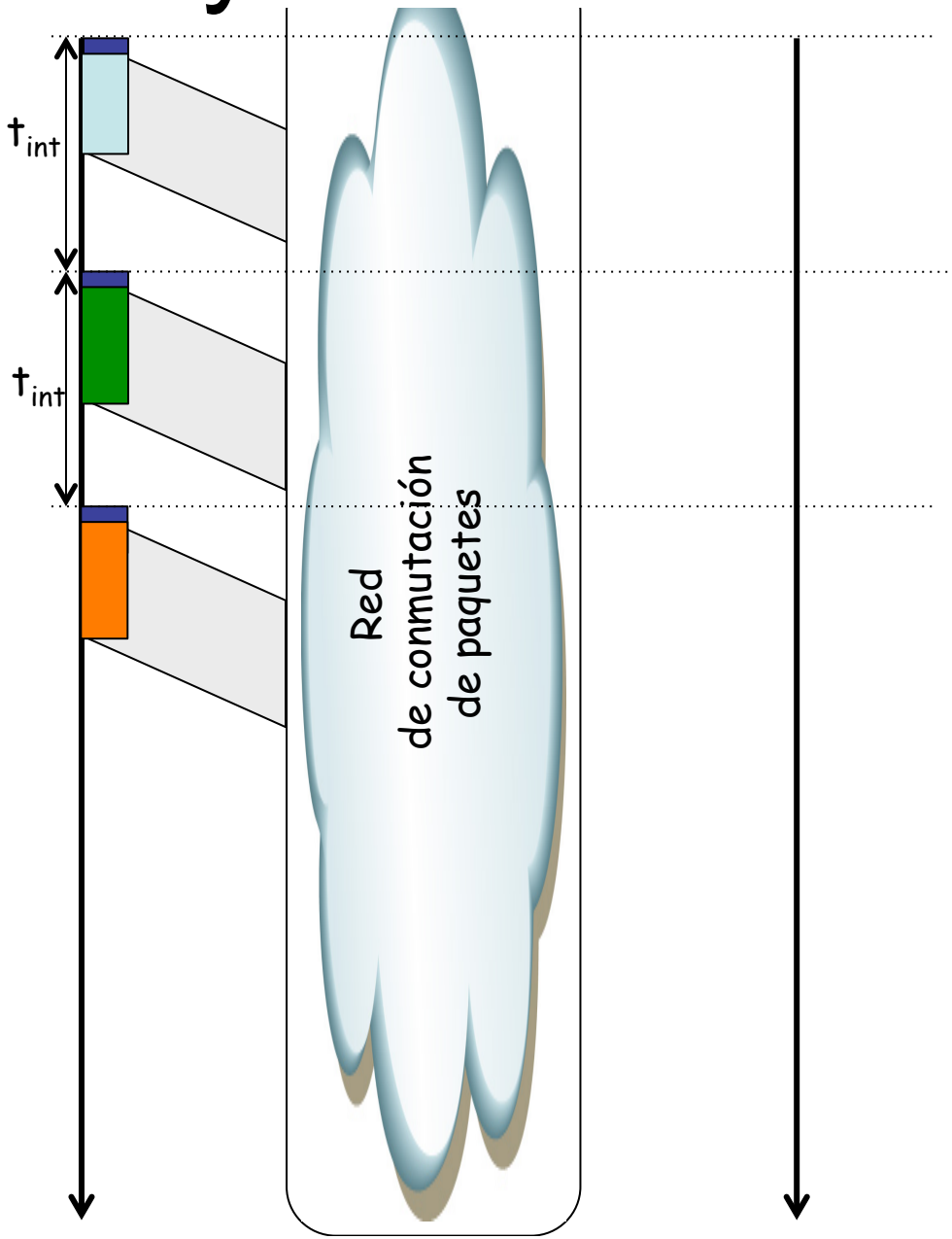


Packet Delay Variation

- Variación en el retardo (*jitter*)

Ejemplo 1

- Paquetes equiespaciados
- (...)

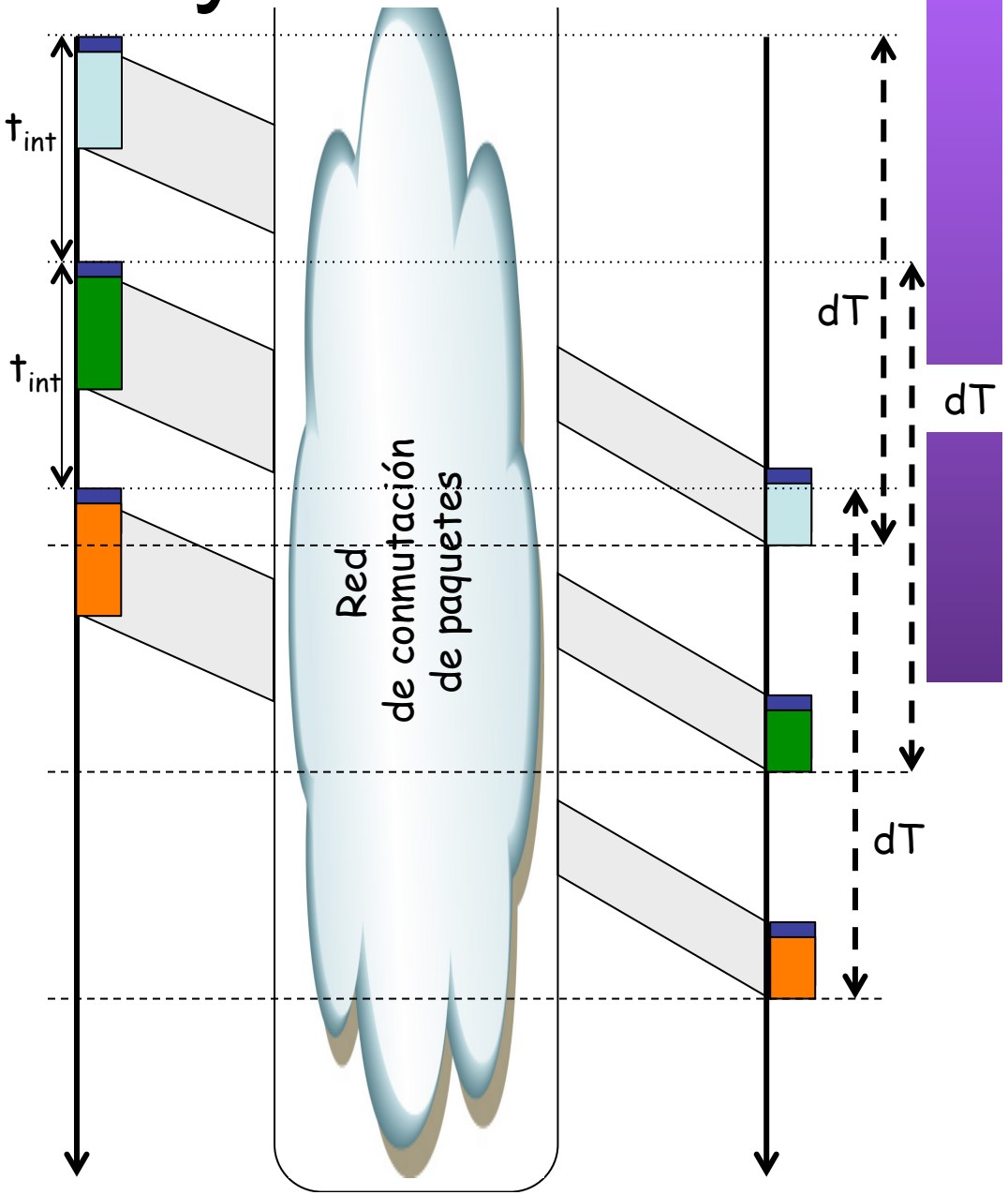


Packet Delay Variation

- Variación en el retardo (*jitter*)

Ejemplo 1

- Paquetes equiespaciados
- Retardo medido entre el tiempo de inicio de envío de primer bit y tiempo de fin de recepción del último bit (dT)
- Todos sufren igual retardo hasta el punto de medida
- (...)

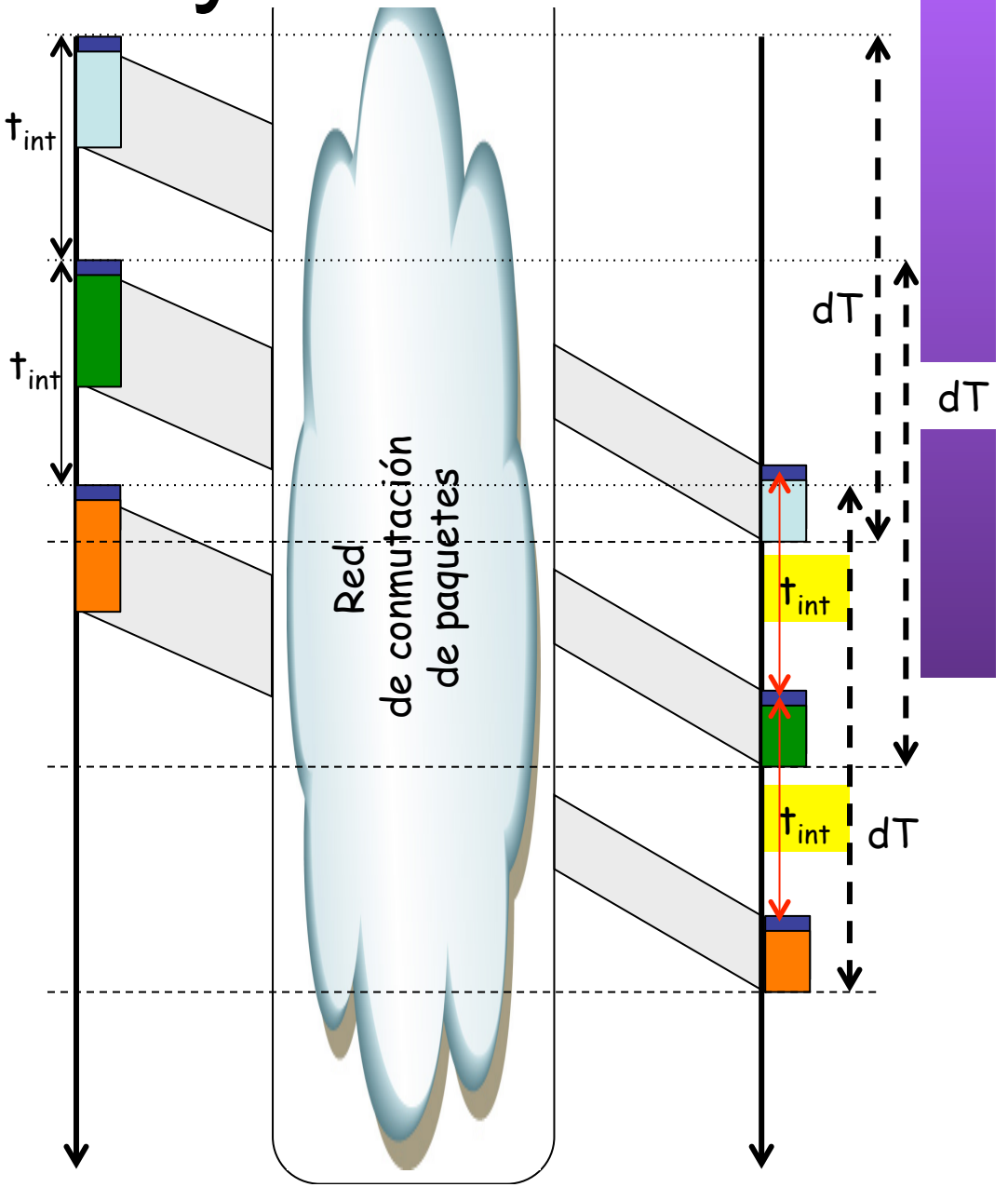


Packet Delay Variation

- Variación en el retardo (*jitter*)

Ejemplo 1

- Paquetes equiespaciados
- Retardo medido entre el tiempo de inicio de envío de primer bit y tiempo de fin de recepción del último bit (dT)
- Todos sufren igual retardo hasta el punto de medida
- En ese otro extremo (o punto de medida) los paquetes están equiespaciados
- No hay variación en el retardo

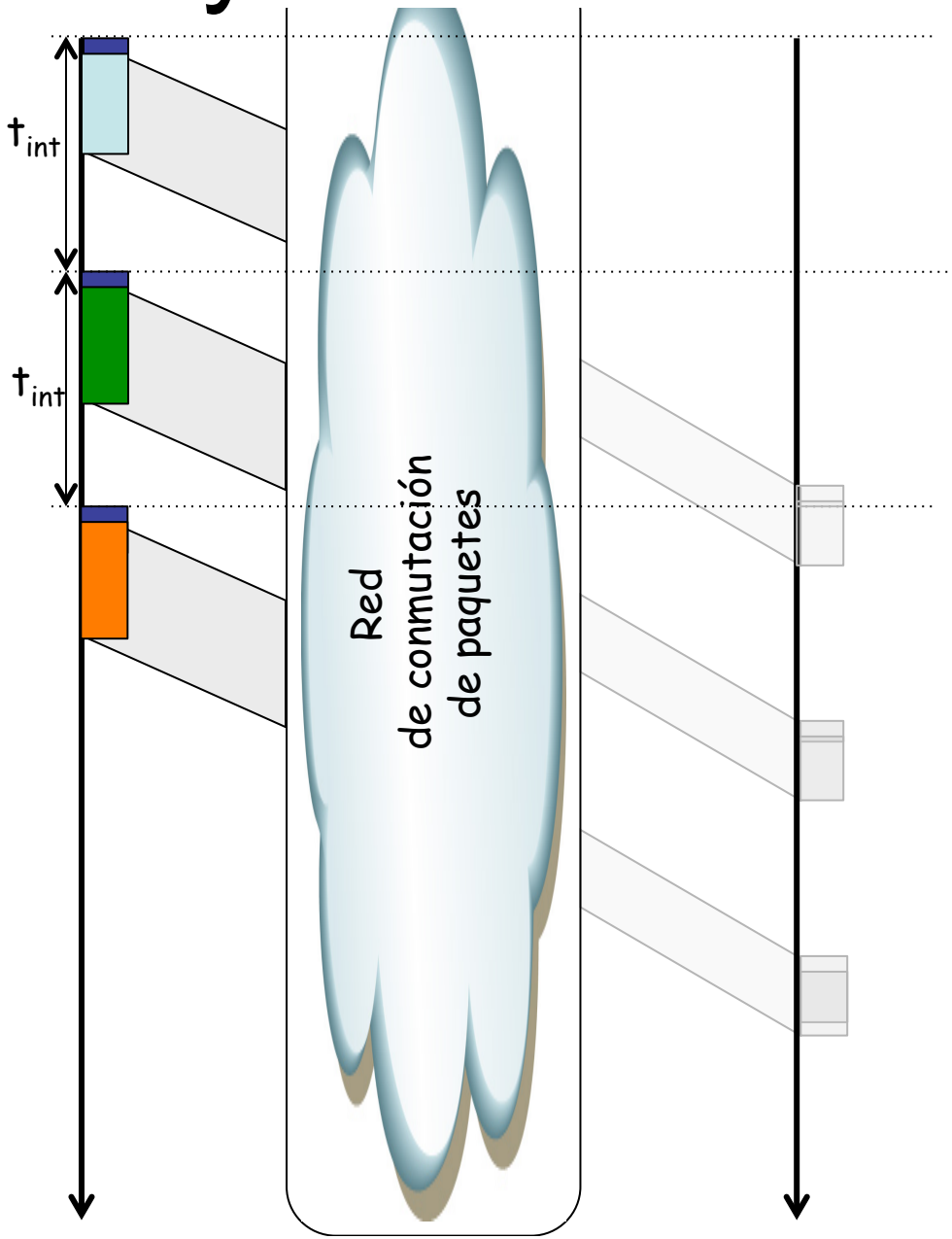


Packet Delay Variation

- Variación en el retardo (*jitter*)

Ejemplo 2

- Paquetes equiespaciados
- (En gris los instantes del ejemplo anterior)
- Sufren diferente retardo (dT_1, dT_2 y dT_3) (...)

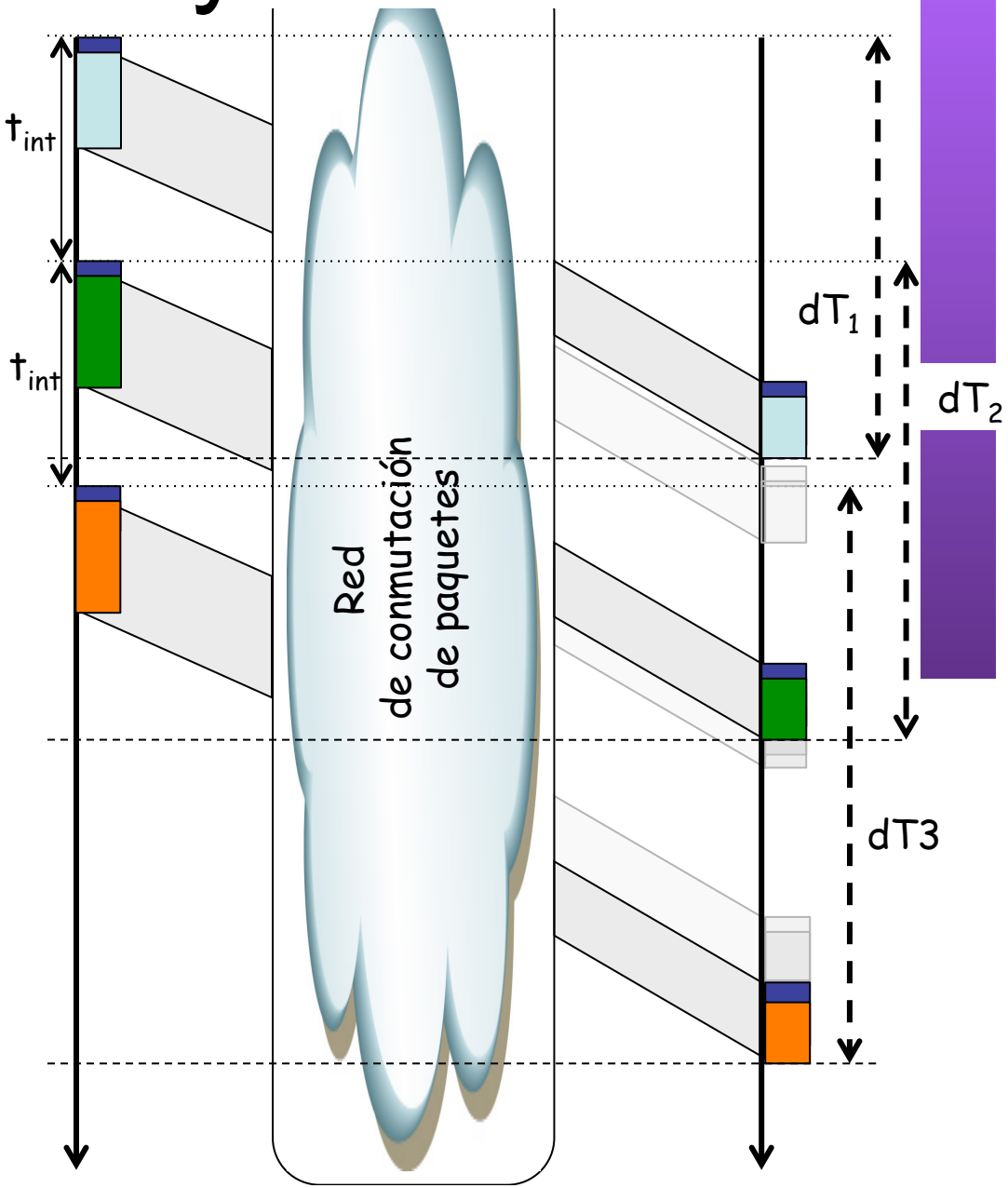


Packet Delay Variation

- Variación en el retardo (*jitter*)

Ejemplo 2

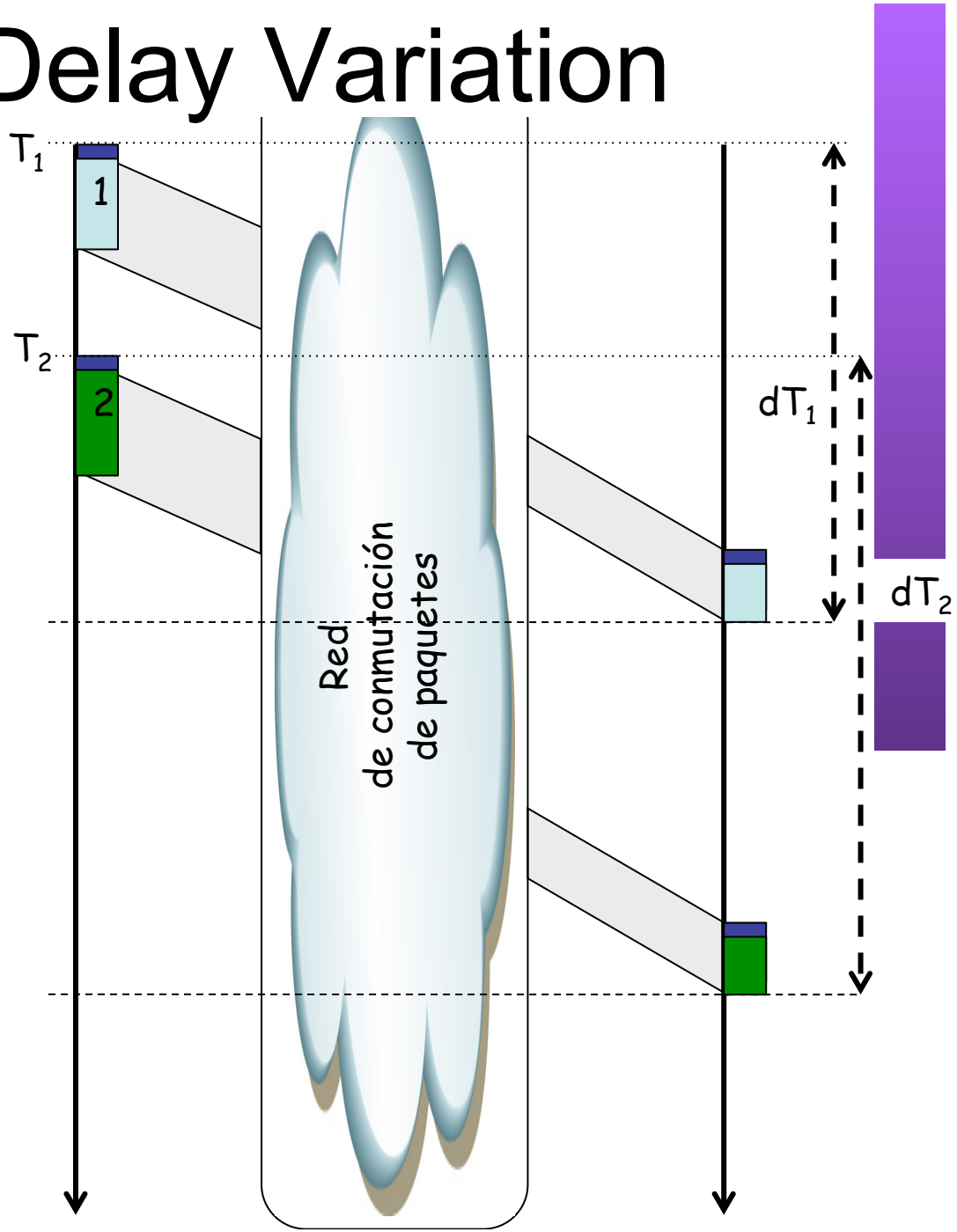
- Paquetes equiespaciados
- (En gris los instantes del ejemplo anterior)
- Sufren diferente retardo (dT_1, dT_2 y dT_3)
- PDV mide la variación en el retardo



Packet Delay Variation

Cálculo

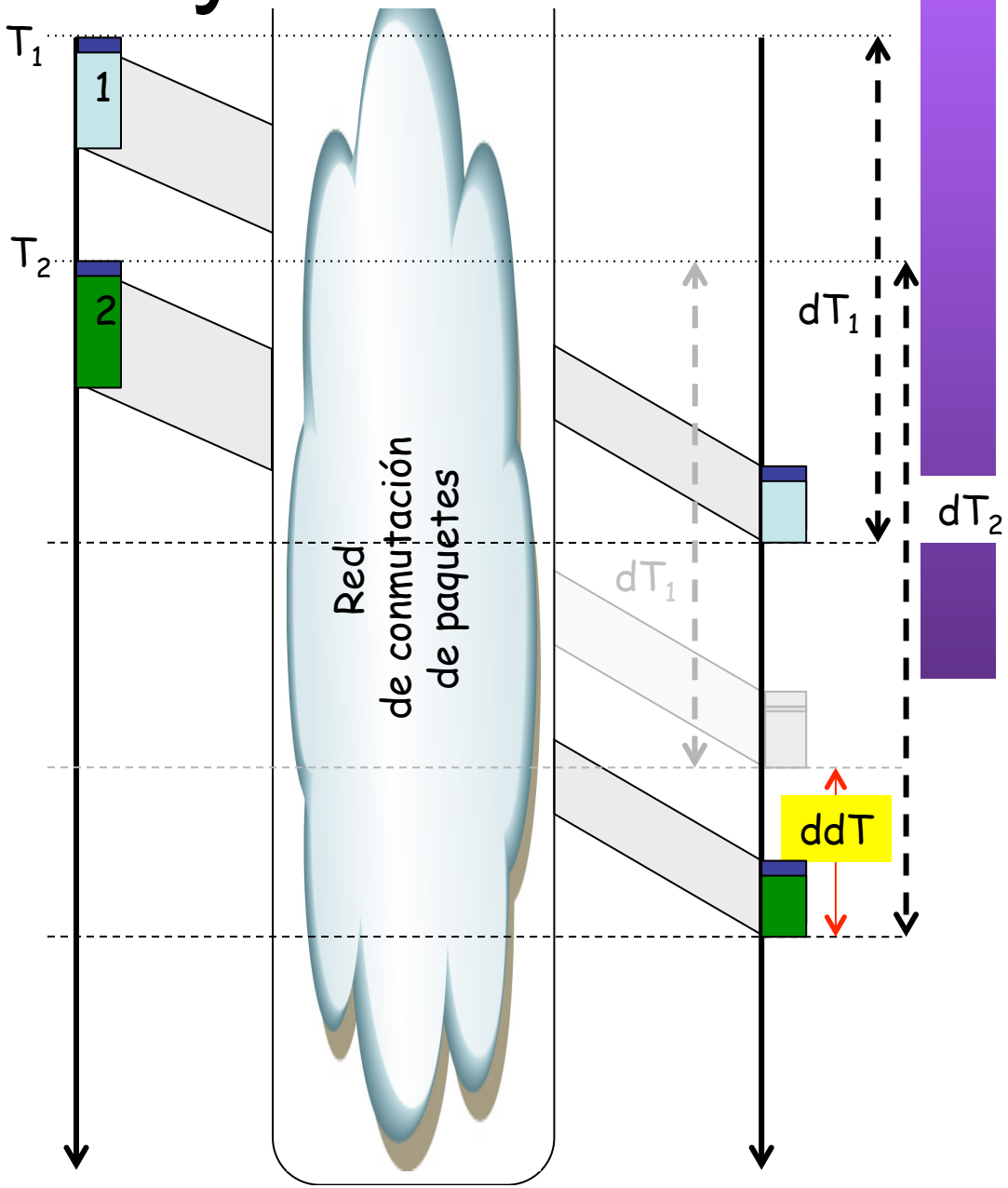
- Dos paquetes (1) y (2)
- Retardos dT_1 y dT_2
- $ddT = dT_2 - dT_1$
- Mide la diferencia entre cuándo ha llegado el segundo paquete y cuándo "debería" haber llegado
- El "debería" sería en el caso de mismo retardo ambos (...)



Packet Delay Variation

Cálculo

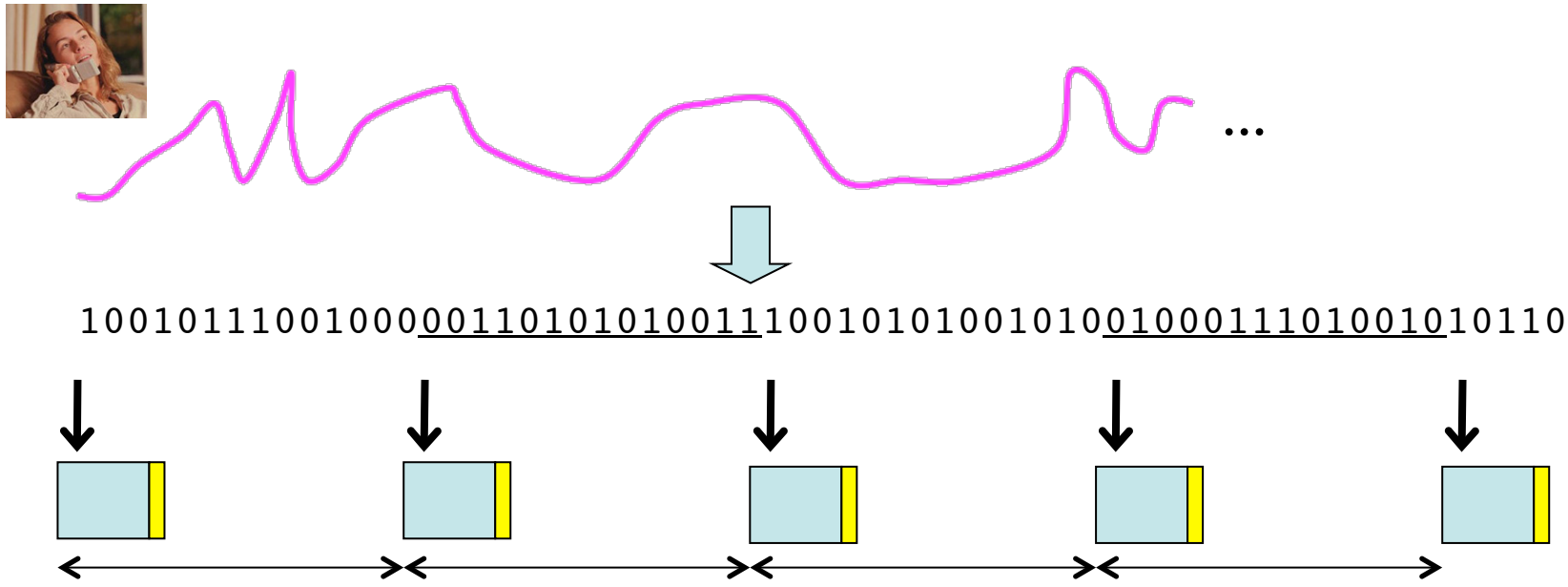
- Dos paquetes (1) y (2)
- Retardos dT_1 y dT_2
- $ddT = dT_2 - dT_1$
- Mide la diferencia entre cuándo ha llegado el segundo paquete y cuándo "debería" haber llegado
- El "debería" sería en el caso de mismo retardo ambos (paquete en gris)
- Diferencia puede ser positiva o negativa (atrasarse o adelantarse)



Efectos del PDV

Ejemplo

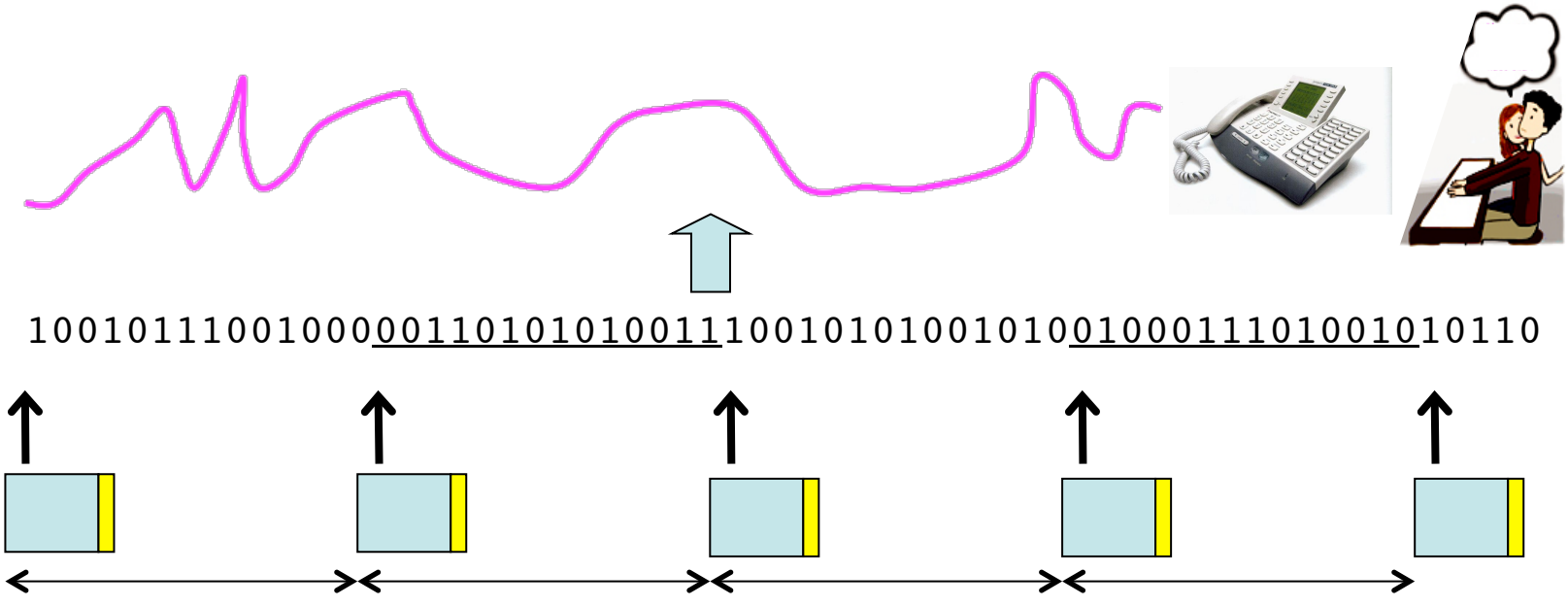
- Codec de voz que genera información digital a tasa constante
- Una vez paquetizada se convierte en paquetes equiespaciados
- (...)



Efectos del PDV

Ejemplo

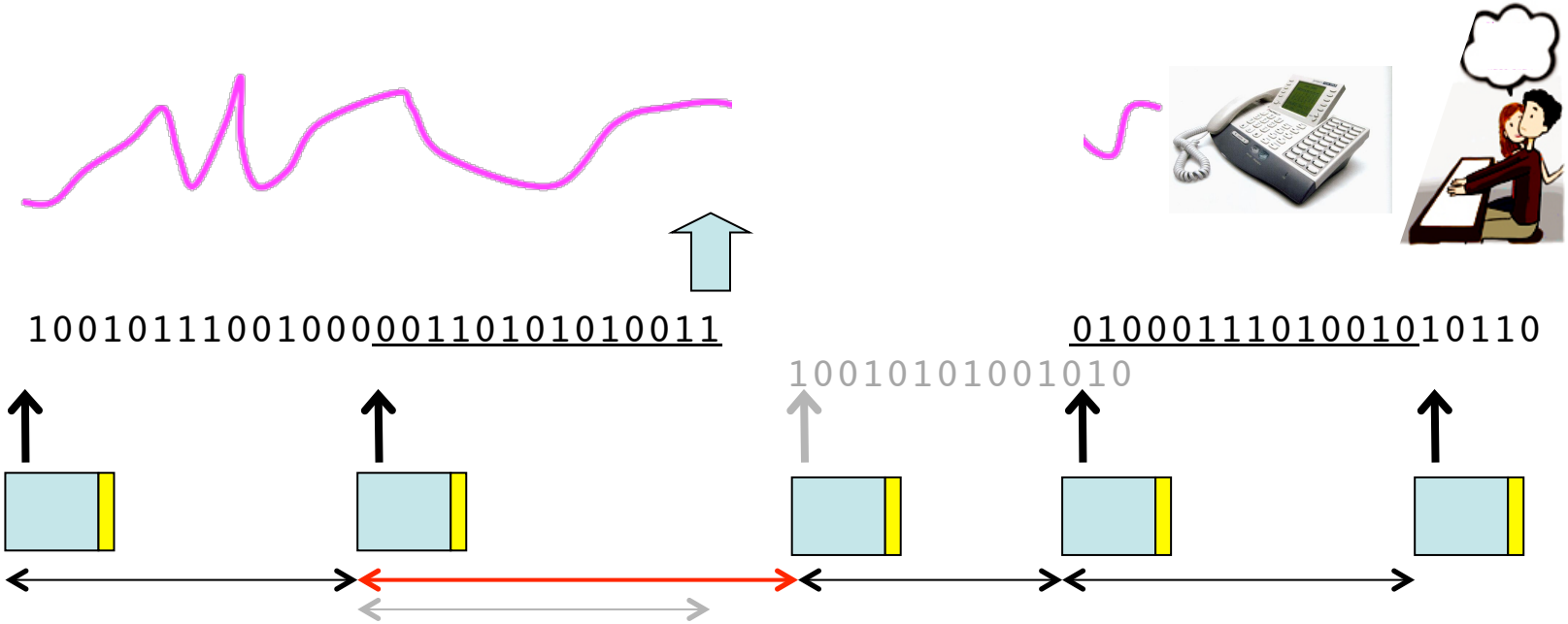
- Codec de voz que genera información digital a tasa constante
- Una vez paquetizada se convierte en paquetes equiespaciados
- En la decodificación se consumen a esa misma tasa
- (...)



Efectos del PDV

Ejemplo

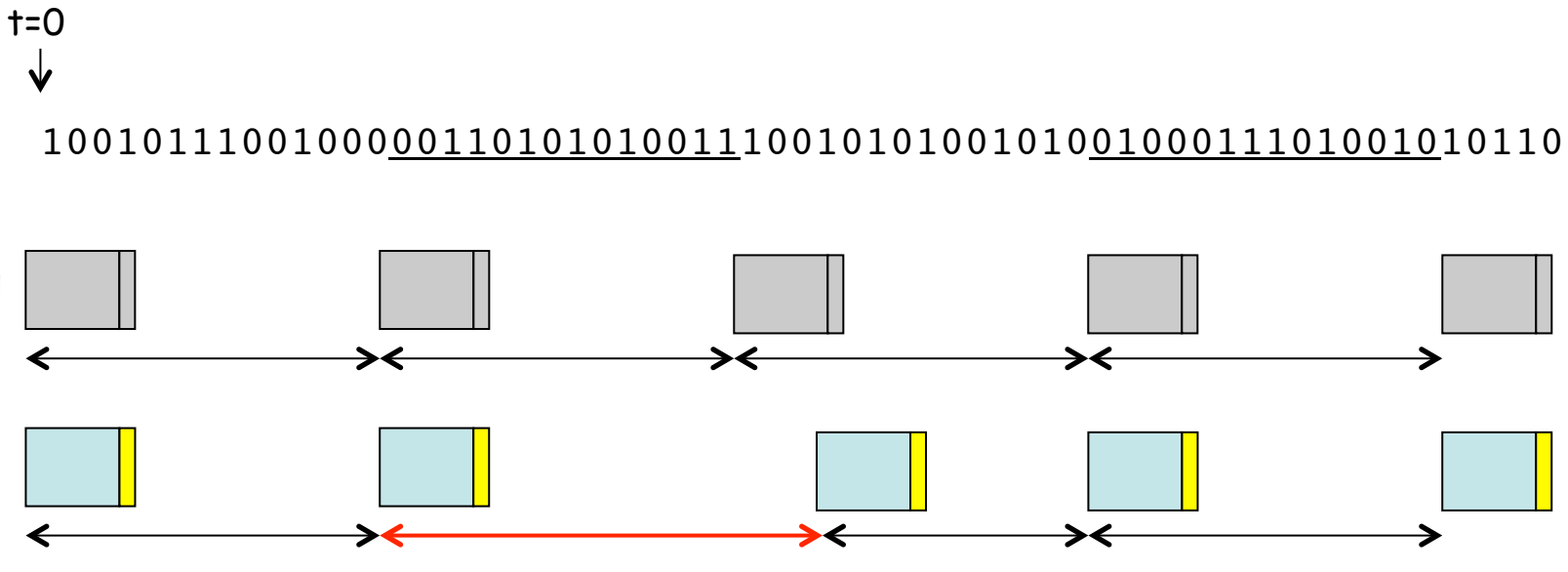
- Codec de voz que genera información digital a tasa constante
- Una vez paquetizada se convierte en paquetes equiespaciados
- En la decodificación se consumen a esa misma tasa
- Un primer paquete sufre un retardo mayor que el anterior y puede que cuando llegue “ya sea tarde”
- Es decir, ya no sirve decodificarlo pues ya se ha producido el corte en la reproducción



Efectos del PDV

Solución

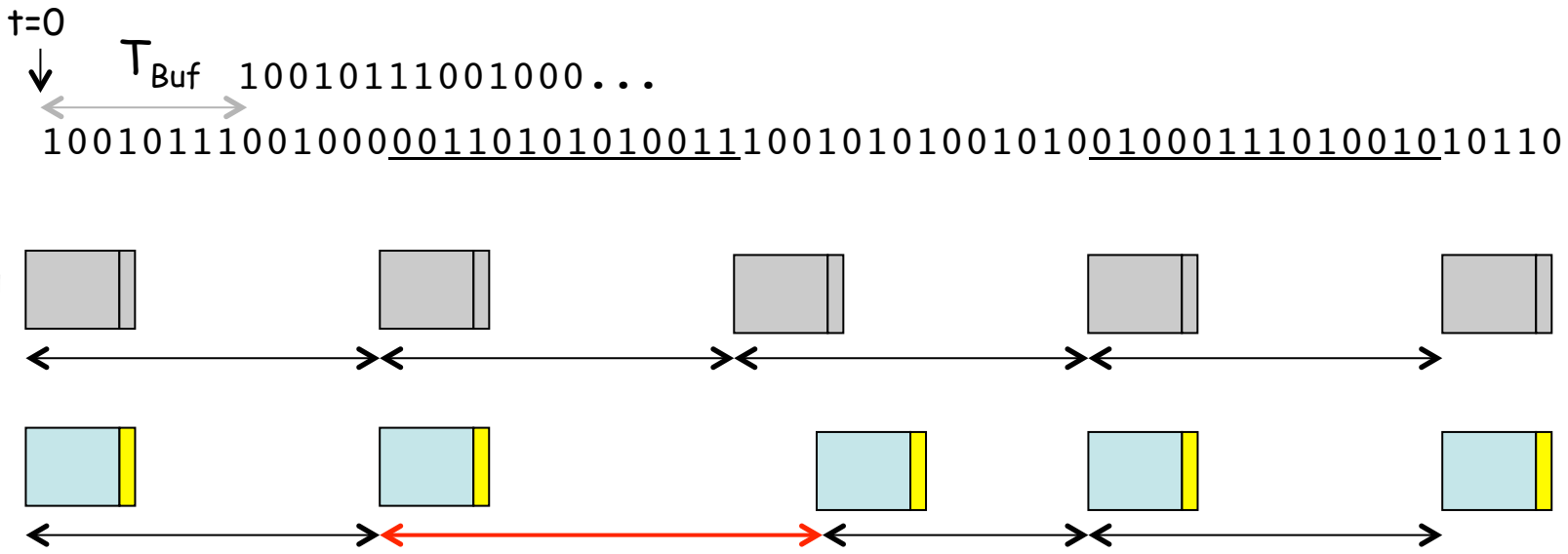
- Retrasar comienzo de la reproducción mediante *buffering* en el cliente
- Supongamos que en t=0 tiene el primer paquete y podría empezar a reproducir
- (...)



Efectos del PDV

Solución

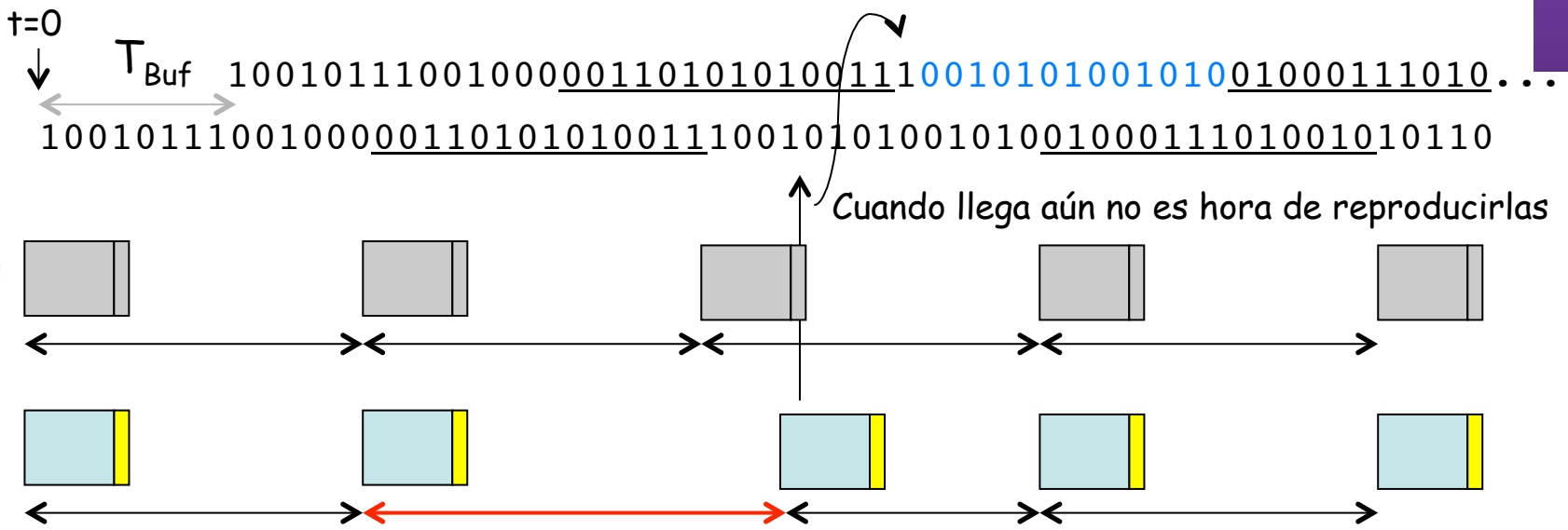
- Retrasar comienzo de la reproducción mediante buffering en el cliente
- Supongamos que en t=0 tiene el primer paquete y podría empezar a reproducir
- Se introduce en memoria durante T_{Buf} (mientras tanto pueden llegar más paquetes, según el tiempo que se desee y lo grande que sea T_{Buf})
- (...)



Efectos del PDV

Solución

- Retrasar comienzo de la reproducción mediante buffering en el cliente
- Supongamos que en t=0 tiene el primer paquete y podría empezar a reproducir
- Se introduce en memoria durante T_{Buf} (mientras tanto pueden llegar más paquetes, según lo grande que sea T_{Buf})
- El paquete muy retrasado entrará en el buffer y aún se estarán reproduciendo muestras de anteriores si su PDV es menor que T_{Buf}

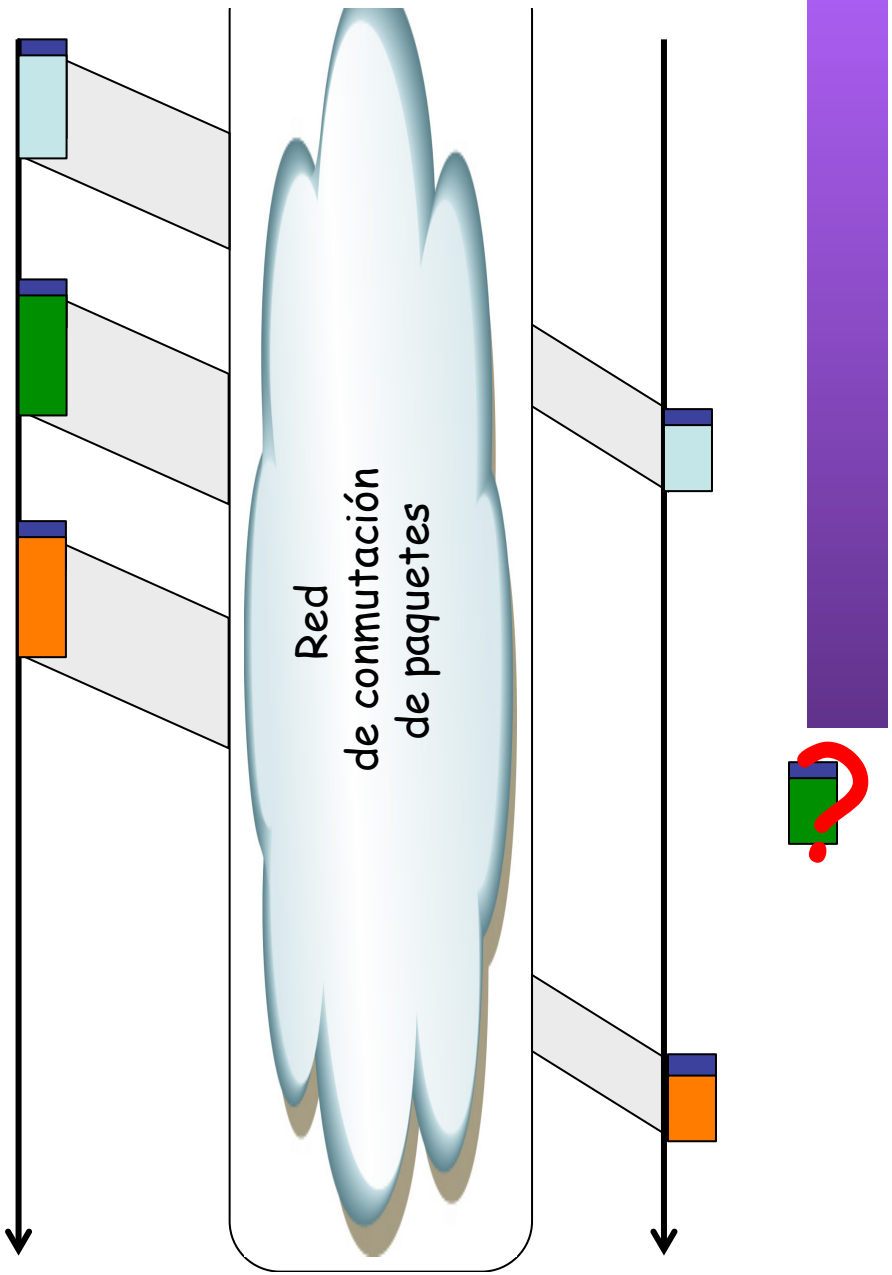


Pérdidas

- Los paquetes podrían no llegar nunca a su destino

Posibles motivos

- (...)



Pérdidas

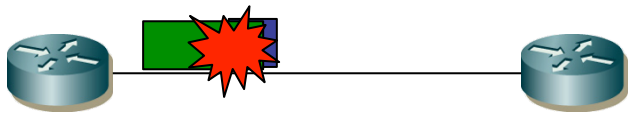
- Los paquetes podrían no llegar nunca a su destino

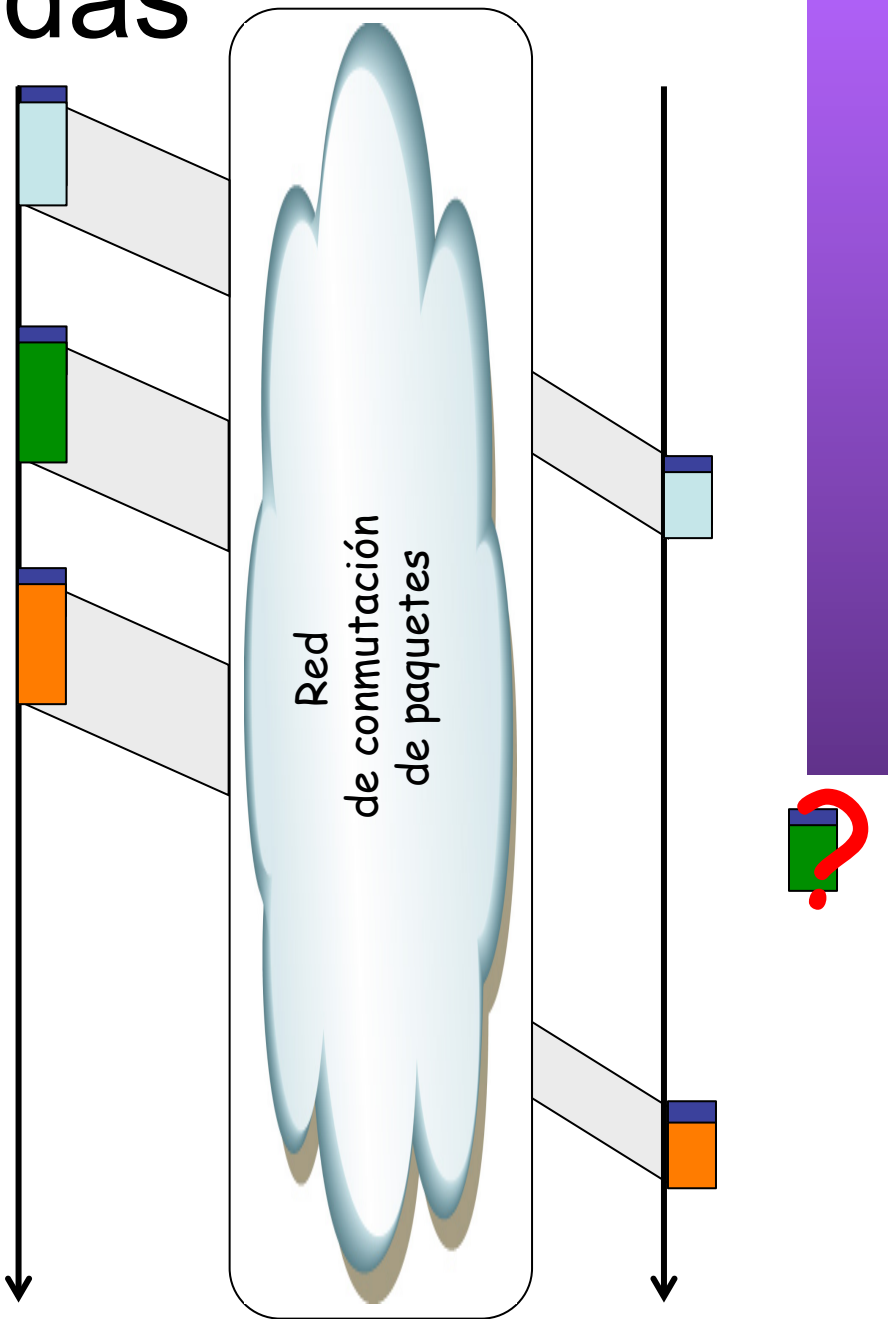
Posibles motivos

- Se corrompió y fue descartado en algún nodo de la red (CRCs)
- BER = Bit Error Rate
- Aproxima a la probabilidad de error de bit p_{err}
- Probabilidad de algún error en un paquete de N bits:

$$p_{epk} = 1 - (1 - p_{err})^N$$

- Asumiendo errores indep. (no ráfagas)
- Sin código "corrector" de errores
- Ejemplo: $p_{err}=10^{-6}$, N=12.000 ➔ ➔ $p_{epk}\approx 10^{-2}$
- (...)



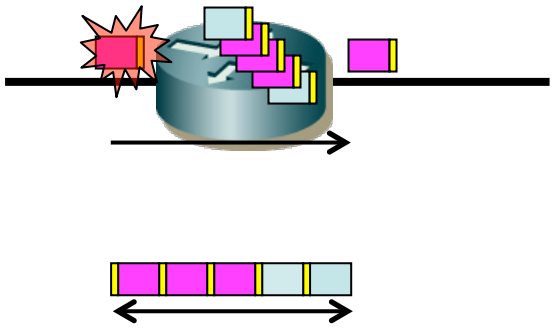


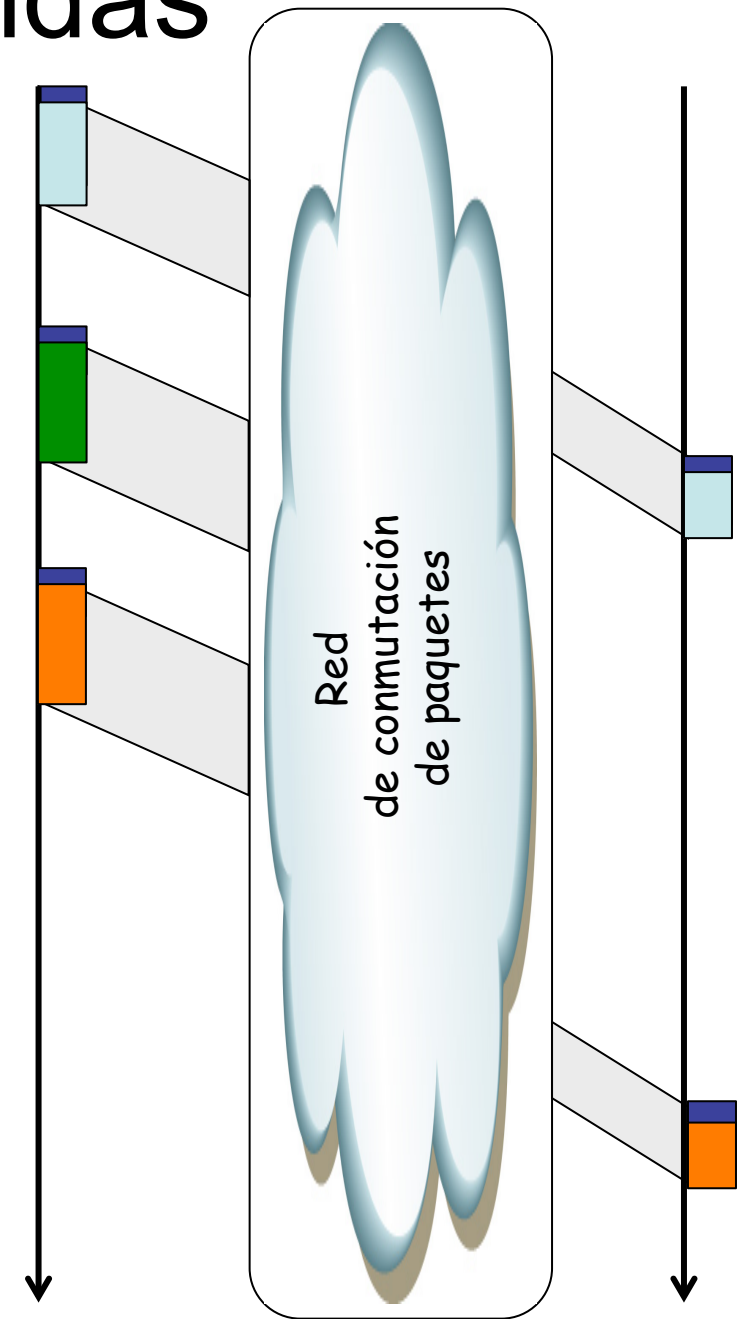
Pérdidas

- Los paquetes podrían no llegar nunca a su destino

Posibles motivos

- Se descartó en un nodo de la red por desbordamiento de buffer
- (...)



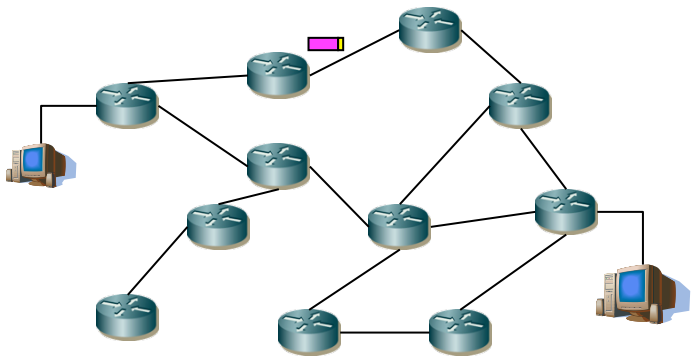


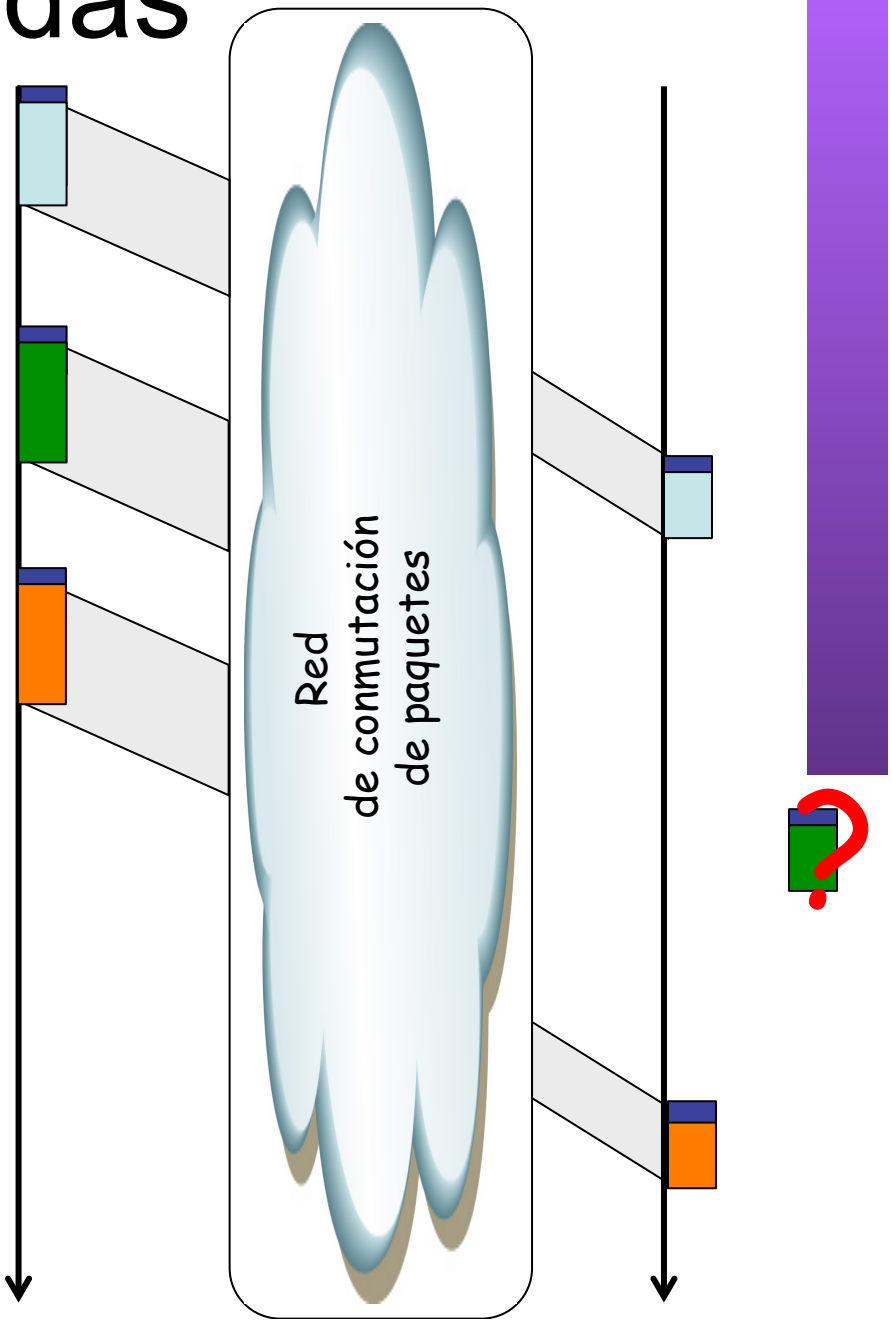
Pérdidas

- Los paquetes podrían no llegar nunca a su destino

Posibles motivos

- Se descartó por exceder el tiempo en la red
- (...)



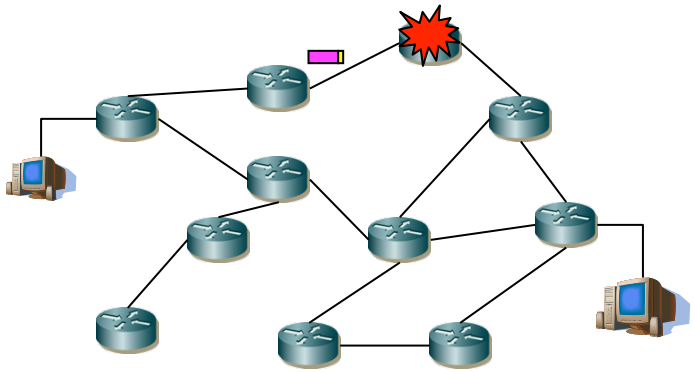


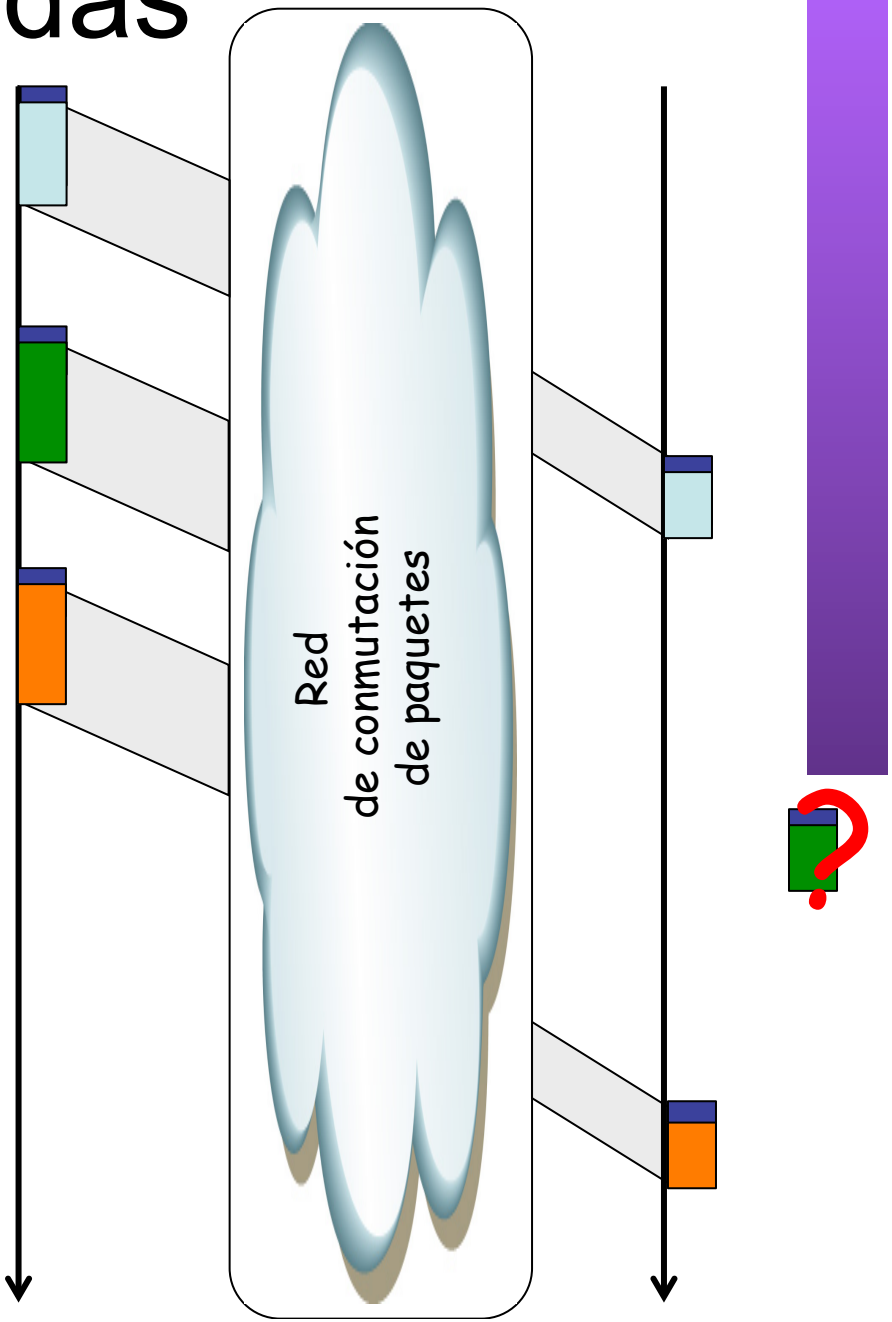
Pérdidas

- Los paquetes podrían no llegar nunca a su destino

Posibles motivos

- Fallo de un elemento de red
- Lleva un tiempo recalcular caminos
- (...)



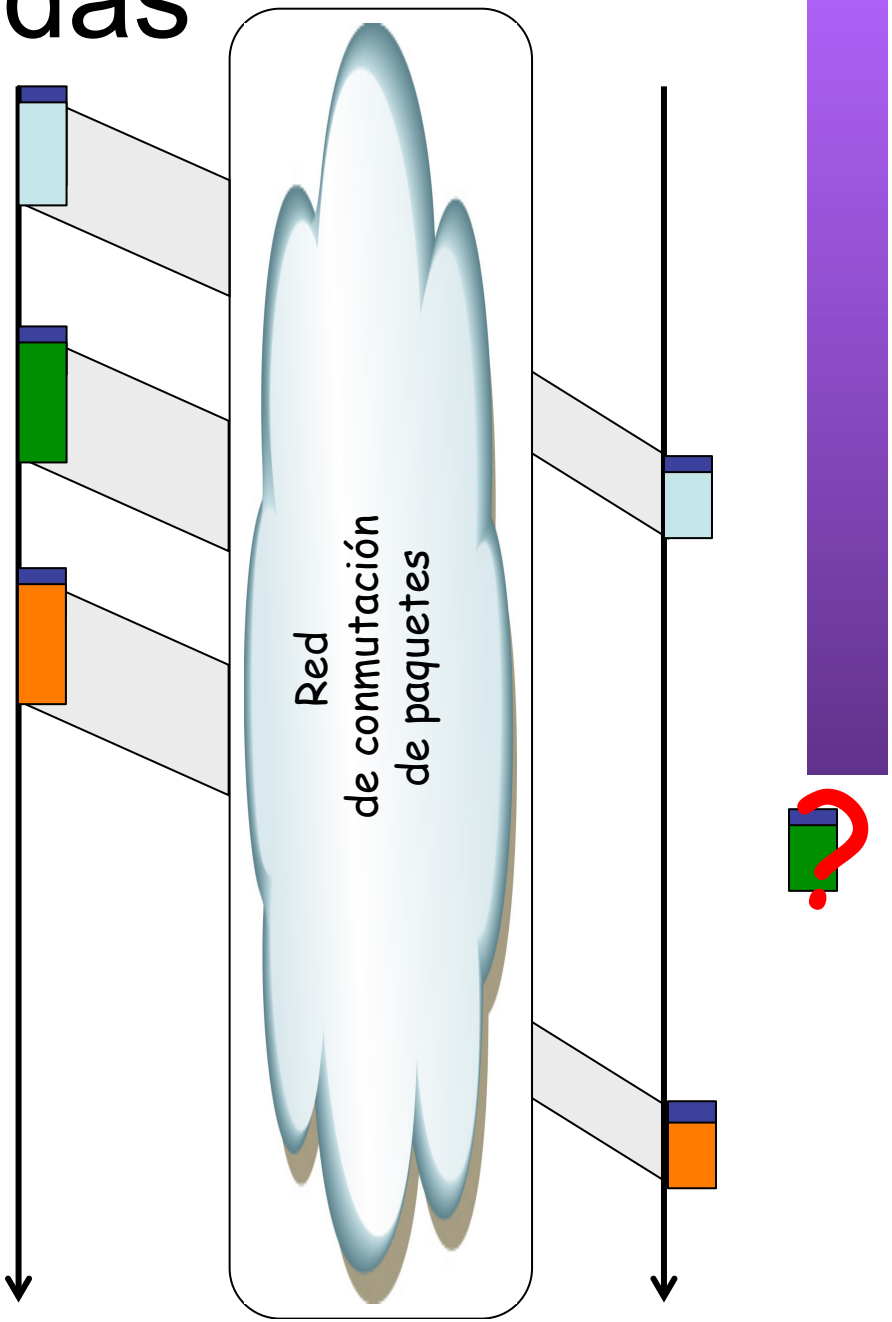


Pérdidas

- Los paquetes podrían no llegar nunca a su destino

Posibles motivos

- Descarte en nodo extremo por desbordamiento de buffer
- Puede ser culpa de la propia aplicación y del tiempo que le lleva procesar los datos recibidos



Ejemplo

Voz y pérdida de paquetes

- Causan cortes y saltos
- Un paquete suele contener en torno a 20ms de muestras de voz
  - Si contiene menos, mayor ratio cabeceras/datos
  - Si contiene más, mayor retardo de formación
- Pérdida de 1 paquete se puede intentar recuperar (interpolación, etc)
- Pérdida de más de 1 paquete crea un corte que se nota claramente

Paquetes retrasados y jitter

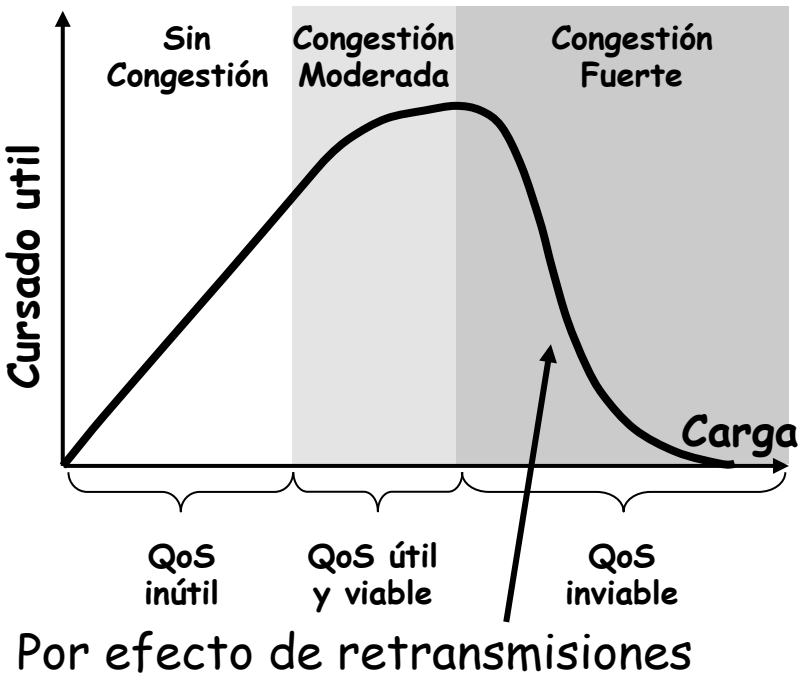
- Si un paquete llega demasiado tarde es equivalente a una pérdida
- Si el retardo general end-to-end es demasiado grande se pierde interactividad
- El jitter (variación en el retardo) se puede recuperar con buffer en el receptor
- Si el jitter es demasiado grande el buffer ha de ser tan grande que de nuevo implica demasiado retardo

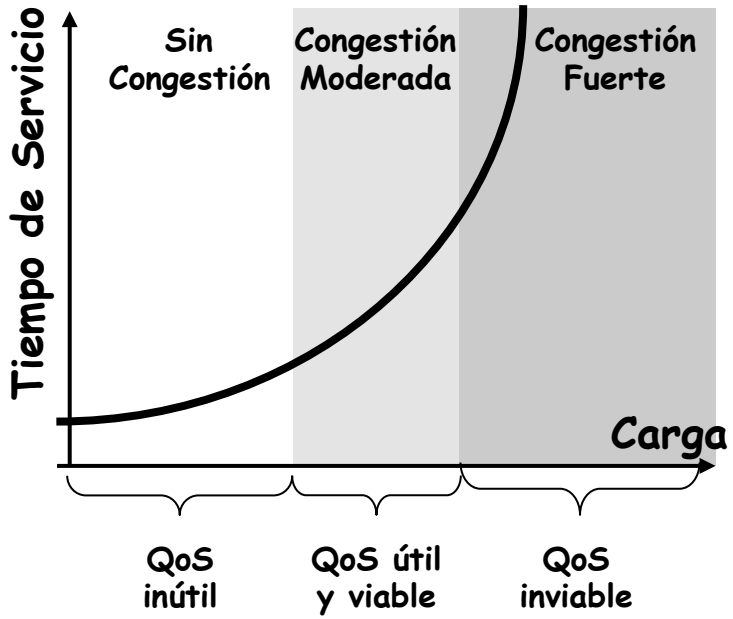
Availability

- **Network availability**
  - Fracción de tiempo en que la conectividad está disponible
  - Puede faltar la conectividad por cortes programador o por fallos
  - Se tiene en cuenta la disponibilidad de cada elemento y se combinan, según estén en serie o en paralelo
  - En serie deben estar todos disponibles (“disponible link 1 Y disponible link 2 Y disponible link 3 ...”)
  - En paralelo debe estar alguno disponible (“disponible link 1 O disponible link 2 O disponible link 3 ...”)
- **Service availability**
  - Fracción de tiempo en que el servicio está disponible dentro de los parámetros de SLA del servicio
  - Puede medirse independiente de la disponibilidad de red en cuyo caso no puede superarla o medirse condicionada a disponibilidad
  - Puede implicar al comportamiento de servidores, según cómo se haya definido el servicio

Congestión y Calidad de Servicio

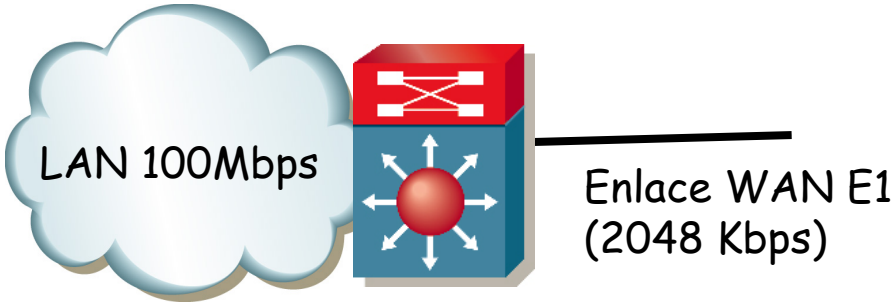
- Congestión: colas llenas
- Fácil dar QoS si nunca hay congestión
- Para dar QoS con congestión
  - Opción 1: Gestionar los recursos ante congestión para dar un trato diferenciado (congestion control/management)
  - Opción 2: Evitar la congestión (congestion avoidance)





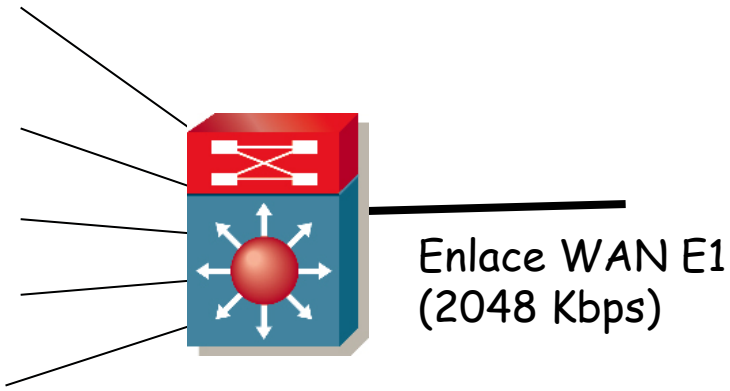
Escenarios con congestión

- Interconexión LAN-to-WAN
  - Tráfico desde LAN de > 10Mbps hacia WAN de < 10Mbps
- (...)



Escenarios con congestión

- Interconexión LAN-to-WAN
  - Tráfico desde LAN de > 10Mbps hacia WAN de < 10Mbps
- Agregación
- Desajuste de velocidades



Solución

- *Throw more bandwidth at the problem !!*
- Así no hay congestión
- No siempre es la solución:
  - No siempre es barato aumentar el BW (ej: acceso, *peering*)
  - Si se le da más BW al usuario habrá mayor demanda
  - Un pico en la demanda degradaría la calidad de servicios sensibles
  - Voz: pequeño BW y bajo retardo pero para lograr bajo retardo puede hacer falta un BW desproporcionado (caro)
  - ¡¡ Congestión por tráfico *scavenger* !! (DoS, worm… bastan 10 PCs para saturar 1GEth)
- Hoy en día ya no es suficiente para un ISP con ofrecer un servicio *Best Effort*
- La tecnología para ofrecer QoS ya es asequible y fiable
- ¡ Pero la ingeniería de estas redes no es sencilla !

Resumen

- Calidad objetiva y subjetiva
- Aplicaciones elásticas e inelásticas
- Retardo, throughput, jitter, pérdidas