

Tema 1-1: Introducción a las técnicas usadas en la calidad de servicio (QoS)

José Manuel Arco Rodríguez

Índice

- ♦ Introducción a la QoS
- ♦ Necesidades del tráfico de usuario
- ♦ Técnicas utilizadas en QoS
 - Control de admisión de conexiones
 - Encaminamiento con QoS
 - Conformado
 - Función Policía
 - Control de congestión
 - Control de flujo
 - Planificadores de tráfico
- ♦ Señalización

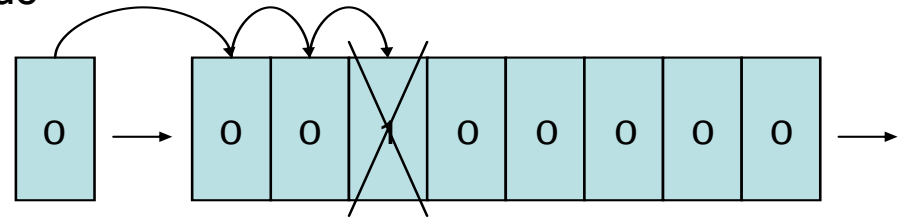
Control de congestión con descarte selectivo

- ♦ Con QoS es improbable que haya congestión, pero puede aparecer por fallos en los routers o enlaces que hace que tráfico se tenga que concentrar por rutas donde puede que no haya los recursos suficientes
- ♦ Este control se hace en la red en la fase de transferencia de datos
- ♦ Previamente, el usuario o la vigilancia de entrada a la red, marcan el tráfico menos prioritario
- ♦ Este control hace que las celdas de más prioridad (CLP=0) reemplacen a las de menos prioridad (CLP=1) en colas congestionadas

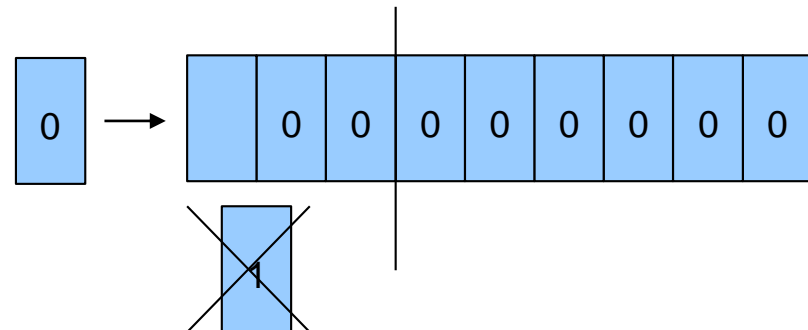
Control de congestión con descarte selectivo

♦ En teoría hay dos posibilidades:

1. Expulsar paquetes menos prioritarios. Implica reordenar la cola que lleva tiempo, por lo que no es empleado



1. No admitir mas paquetes, a partir de cierto umbral de llenado



Índice

- ♦ Introducción a la QoS
- ♦ Necesidades del tráfico de usuario
- ♦ Técnicas utilizadas en QoS
 - Control de admisión de conexiones
 - Encaminamiento con QoS
 - Conformado
 - Función Policía
 - Control de congestión
 - Control de flujo
 - Planificadores de tráfico
- ♦ Señalización

Técnicas de control de flujo

- ◆ Estas técnicas tratan de evitar la congestión influyendo en la velocidad del emisor
- ◆ En general, no son adecuadas para el tráfico interactivo ya que puede implicar bajar la velocidad más de lo que fuente puede necesitar
- ◆ El emisor cambia la velocidad cuando se le notifica de que hay congestión
- ◆ Hay dos opciones [korose] de avisos
 - Implícito, de forma indirecta, eg usado por TCP cuando se pierde un paquete y vence el temporizador de reenvío
 - Explícito, el nodo congestionado señala del problema a la fuente

Técnicas de control de flujo

- ♦ El control explícito puede ser a su vez:
 - Aviso en ambas direcciones (usado en FR)
 - Hacia el receptor, en el paquete que está sufriendo la congestión, recordar que el emisor puede controlar la velocidad del emisor
 - Hacia el receptor, lo más efectivo, pero hay que esperar a que llegue tráfico en esa dirección
 - Aviso en una dirección, hacia el receptor (usado en ATM)
 - Basado en tasa (visto a continuación)

Control de flujo basado en tasa

- ♦ Utilizado en conexiones ATM ABR (Avaliable Bit Rate) **para datos**
- ♦ La idea es que el emisor transmite a la velocidad que le indique la red en cada momento
 - Esta velocidad, ante congestión de la red, puede bajar o incluso ser cero en algún momento, que implica un retardo adicional imprevisible
 - Por esto no es apropiada para tráfico interactivo
- ♦ Por conexión ABR, cada k células de datos, se envía una célula especial de control RM (Resorce Management)

Control de flujo basado en tasa

- ♦ Esta célula viaja por el mismo camino que los datos y es devuelta al emisor, aunque es tratada de forma diferente
- ♦ El algoritmo es fiable, el emisor sabe cuando deben regresar las células RM
 - Cuando se pierde una, se reduce la velocidad

Control de flujo basado en tasa (Cont.)

- ♦ Cada conexión ABR transmite a una velocidad menor o igual a la *velocidad de transmisión actual* ACR (Actual Cell Rate)
 - La ACR depende del emisor y en último extremo de lo congestionada que esté la red
 - ACR es fijada por la red
 - ACR que estará comprendida entre la MCR (Minimum Cell Rate, mínima) y PCR (Peak Cell Rate, máxima)
- ♦ Cada célula RM lleva la velocidad a la que el emisor quiere transmitir, la *tasa explícita* ER (Explicit Rate), en células por segundo

Control de flujo basado en tasa (Cont.)

- ♦ Por los conmutadores por donde va pasando la célula RM, pueden bajar la ER si hay congestión, nunca subirla
- ♦ También el receptor puede bajar la velocidad
- ♦ La reducción puede producirse en el trayecto de ida o en el de vuelta
- ♦ Cuando el emisor recibe la célula RM ajusta la ACR a la velocidad ER recibida haciendo $ACR=ER$

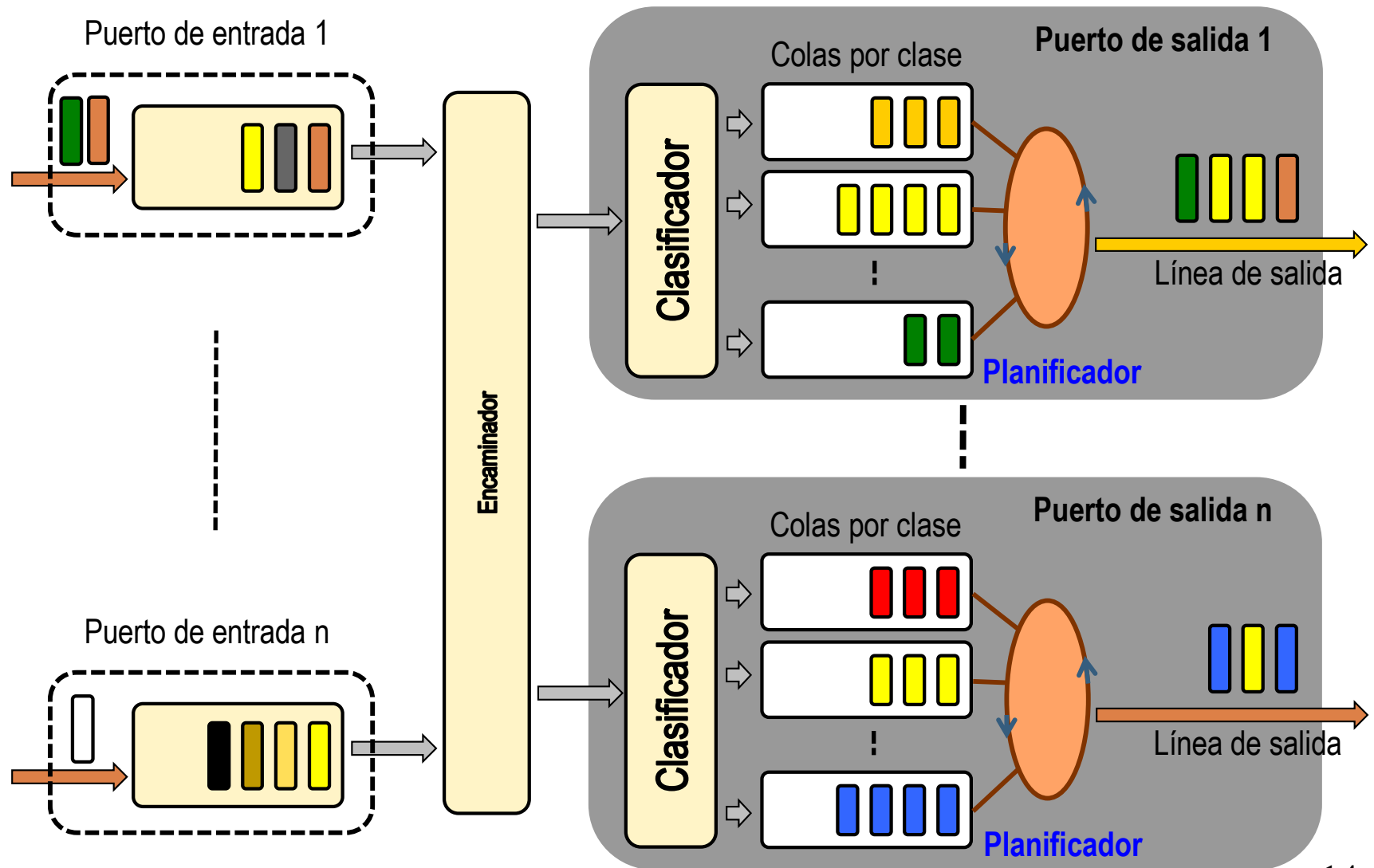
Índice

- ♦ Introducción a la QoS
- ♦ Necesidades del tráfico de usuario
- ♦ Técnicas utilizadas en QoS
 - Control de admisión de conexiones
 - Encaminamiento con QoS
 - Conformado
 - Función Policía
 - Control de congestión
 - Control de flujo
 - Gestores de colas
 - Planificadores de tráfico
- ♦ Señalización

Planificadores de tráfico [kurose 623]

- ♦ El principio general de un planificador es controlar el uso de un recurso compartido, eg, en un SO la CPU
- ♦ El planificador se encarga de asignar un peso a cada flujo o clase de tráfico
- ♦ Tiene que decidir cuál es el siguiente paquete que debe transmitirse sabiendo el peso de flujo
- ♦ Se asocia un planificador a cada línea de salida (puerto) del router
- ♦ No está estandarizado, por tanto, es un campo abierto a la competencia entre fabricantes

Planificadores de tráfico



Planificador FCFS (first-come first-served)

- ♦ Planificación por defecto (o no planificación): los paquetes se sirven en el mismo orden de llegada (FIFO)
 - Muy sencillo y fácil de implementar (solo requiere un buffer de almacenamiento)
 - No distingue entre paquetes por lo que el servicio que proporciona es del tipo 'best-effort'
 - Inconvenientes:
 - En caso de congestión todos los paquetes 'sufren' por igual (resulta ineficiente ya que no todos los paquetes son igual de 'importantes')
 - Favorece el tráfico UDP frente a TCP (UDP no reacciona a las pérdidas producidas mientras que TCP si lo hace reduciendo su tasa)

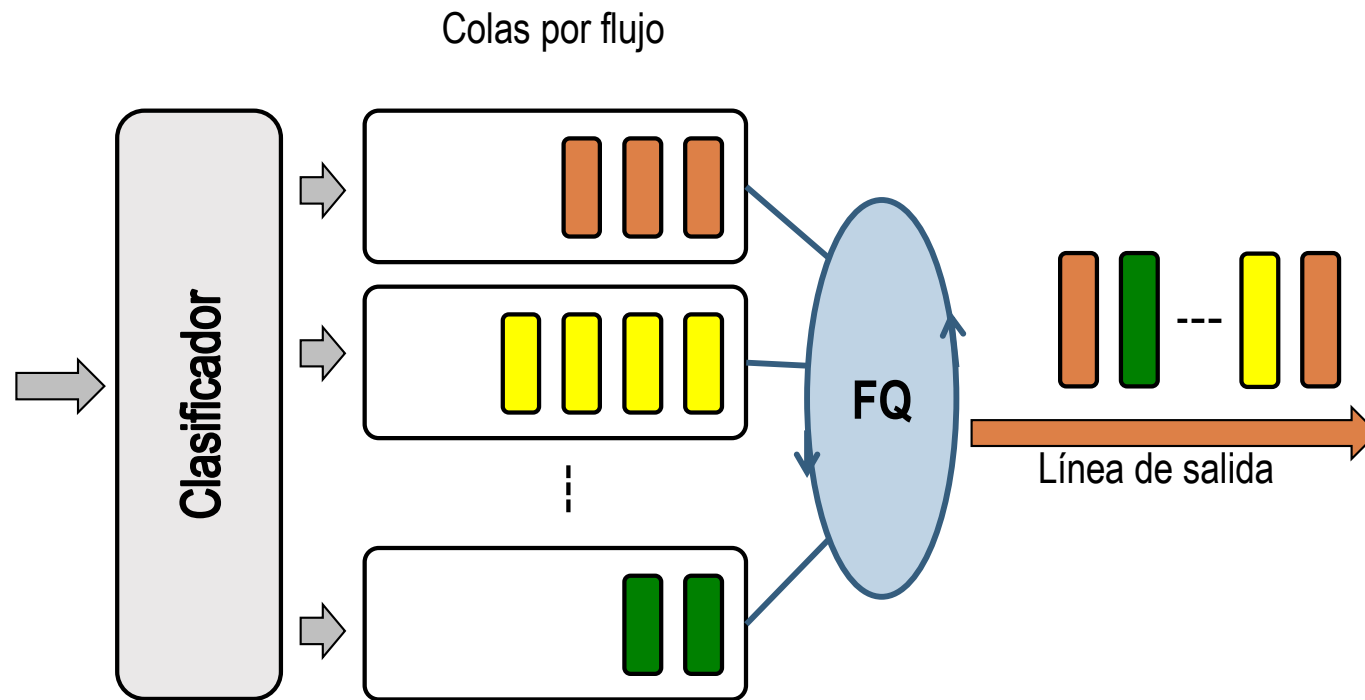
Planificador de prioridad (priority queueing PQ)

- ♦ Visto en la P2-1-2
- ♦ Las colas tienen una prioridad fija
- ♦ Se sirve tráfico de la más prioritaria hasta quedar vacía
- ♦ Después se continua sirviendo la siguiente más prioritaria
- ♦ Se sirve una cola si todas las más prioritarias están vacías
- ♦ Ventajas
 - Sencillo
- ♦ Inconvenientes
 - Inanición
- ♦ Para evitar la inanición
 - Debe haber un control de tráfico (CAC previo, o vigilancia, ver P2-1-3)
 - Se aplica a tráfico con bajo retardo y velocidad (VoIP)

Planificador Round Robin (RR)

- ♦ También llamado Fair Queuing
- ♦ Se sirven las colas de forma circular, a un paquete por cola
- ♦ Las colas sin tráfico se saltan
- ♦ Es sencillo
- ♦ Con tamaños aleatorios de paquetes no se consigue una asignación precisa de ancho de banda y de jitter
- ♦ Inconvenientes:
 - No permite priorizar a una clase (flujo) frente a otras
 - Es sensible al tamaño de los paquetes. El planificador selecciona paquetes completos por lo que favorece a los flujos con paquetes de mayor tamaño

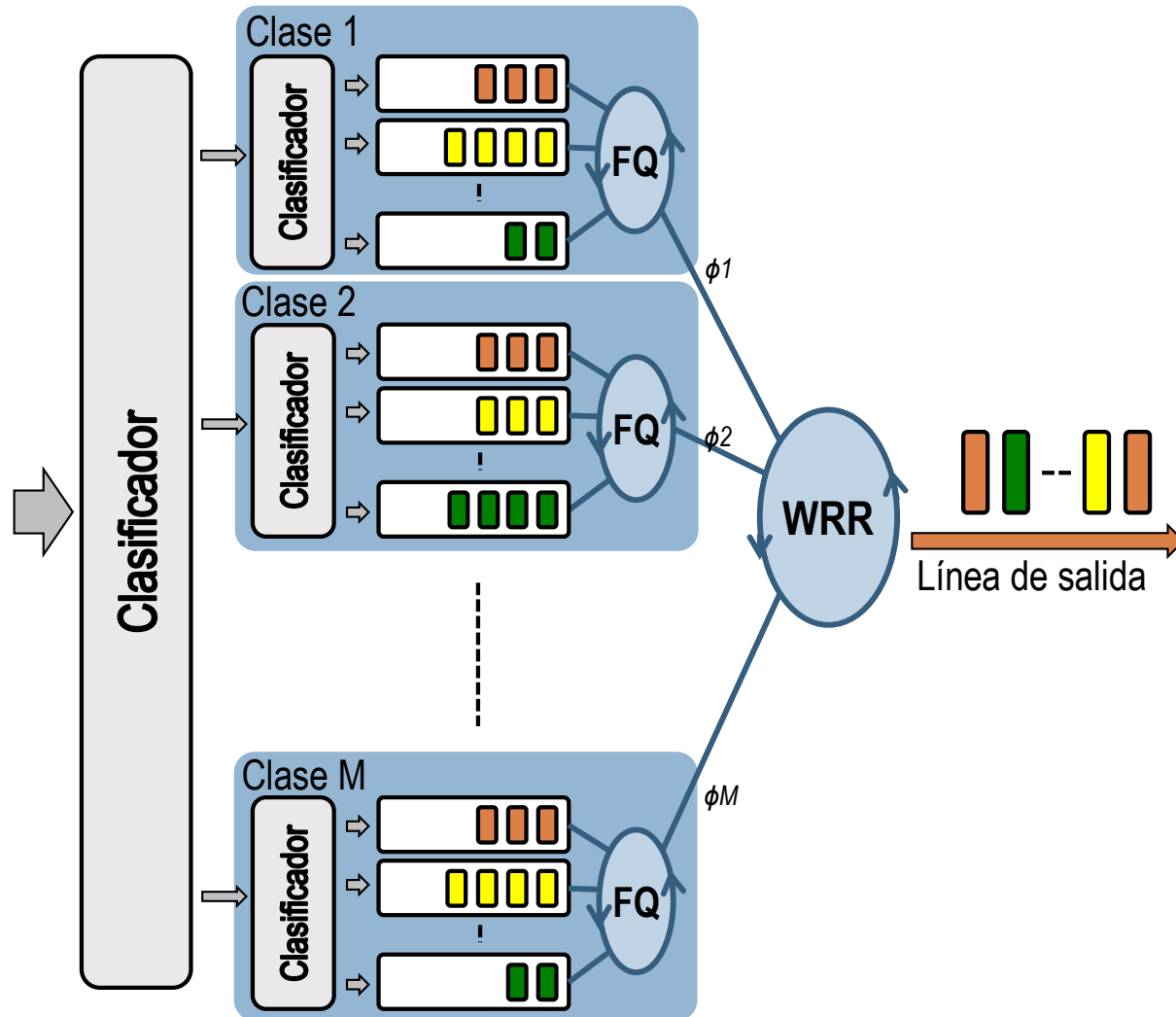
Planificador Round Robin (RR)



Weighted Round Robin (WRR)

- ♦ Es una variación del RR que asigna un peso a la cola que indica el número de paquetes que se servirán cuando le llegue su turno
- ♦ Trata de priorizar unos flujos frente a otros (también conocido como Class-Based Queuing, CBQ, y viene implementado en Linux)
- ♦ Define una arquitectura en dos (o mas) niveles:
 - Los flujos de entrada se agrupan en 'M' clases con su peso
 - Dentro de cada clase los flujos individuales se gestionan mediante colas FQ (todos iguales)
- ♦ Inconvenientes:
 - Resulta muy complejo de gestionar si M es grande.
 - Sigue siendo sensible al tamaño de los paquetes de cada flujo
 - WRR no ofrece bajo retardo ni jitter

Weighted Round Robin (WRR)



Planificador WFQ (Weight Fair Queuing)

- ♦ Hay M clases (o colas) y la suma de los pesos Φ_i es 1
- ♦ A la llegada de un paquete, se le calcula un tiempo virtual de finalización o Time Stamp (TS), (función del peso de la cola y tamaño del paquete) y se encola
 - Eg, para el paquete K ésimo de la cola i, el TS será

$$TS^k = \frac{L_i^k}{f_i Vtx}$$

- ♦ A la salida, el planificador transmite el paquete que tenga el tiempo virtual de finalización menor (si coinciden dos se mandan en cualquier orden)
- ♦ Las conexiones sin QoS van a una misma cola con bajo peso

Ejemplo WFQ

- ◆ Tenemos un sistema con tres clases A, B y C de pesos (A=30%, B=60% y C=10%), y un paquete de tamaño $L_{\text{paq}}=600$ B en espera en cada clase. La línea de salida tiene una velocidad $V_{\text{tx}}= 1$ Mbps

- ◆
 - $t_F(A) = 600 \cdot 8 / (0,3 \cdot 10^6) = 16 \text{ ms}$
 - $t_F(B) = 600 \cdot 8 / (0,6 \cdot 10^6) = 8 \text{ ms}$
 - $t_F(C) = 600 \cdot 8 / (0,1 \cdot 10^6) = 48 \text{ ms}$Por tanto el orden de salida de los paquetes será: B, A, C.

El tiempo real de transmisión de un paquete será: $t_{\text{TX}} = 600 \cdot 8 / 10^6 = 4,8 \text{ ms}$

Y los instantes de tiempo reales en que comienza a transmitirse cada paquete: $t=0$ ms (B), $t=4,8$ ms (A) y $t=9,6$ ms (C)

Planificador WFQ (Cont.)

- ◆ Para calcular el TS de un paquete k hay que tener en cuenta el valor del TS del paquete anterior, k-1

$$TS_i^k = TS_i^{k-1} + \frac{L_i^k}{f_i V_{tx}}$$

- ◆ Se penaliza los paquetes de mayor longitud, ocupan más tiempo la línea y además son los paquetes de datos
- ◆ Esta fórmula tiene un problema cuando no hay paquete previo o hace mucho tiempo que se envió el paquete anterior
- ◆ Veamos el problema en la siguiente transparencia

WFQ (Cont.)

♦ Ejemplo

- Una conexión deja de tener tráfico cuando las etiquetas TS del sistema valía sobre 100
 - Vuelve a tener tráfico cuando las etiquetas valen sobre 1000
 - Si toma como referencia el valor del paquete anterior (100), hará esperar al resto de conexiones hasta que el valor de su TS sea 1000
- ♦ Para solucionar el problema se utiliza la función $v(t)$ “tiempo virtual del planificador”
- Es una función que monitoriza el funcionamiento del sistema

WFQ (Cont.)

- En el ejemplo anterior, cuando la cola vuelve a tener tráfico $v(t)$ valdría sobre 1000,
- ♦ Ahora para calcular el valor del TS, primero calculamos el IT (Initial Time) como el máximo entre la última etiqueta de la cola y la $v(t)$ en ese instante y luego sumamos el otro término

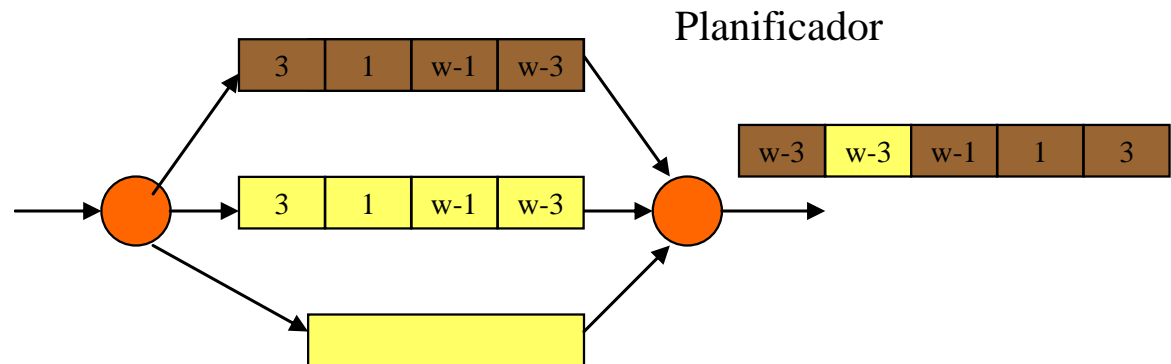
$$TS_i^k = IT_i^k + \frac{L_i^k}{\phi_i V t x}$$

$$IT_i^k = \max(TS_i^{k-1}, v(t_a))$$

- $IT = \max(100, 1000) = 100$, de esta forma no hace esperar a las demás colas
- ♦ $V(t)$ es compleja de calcular, pero hay que tener en cuenta que
 - Toma un valor inferior a los últimos TS calculados por el planificador
 - Para una cola que tenga siempre tráfico para mandar, $TS > v(t)$, es decir no hace saber su valor (dato usado en los ejercicios)

Implementación de WFQ

- ◆ Cuando no hay tráfico las variables se ponen a 0
- ◆ Para evitar desbordamiento en la asignación de TS, se dan cíclicamente, por ejemplo $TS \bmod W; W=2^n-1$
 - Esto tiene un problema que ilustra en la figura, ya que al dar la vuelta al contador, las TS son más bajas y adelantan al tráfico que se iba a enviar antes



- ◆ Solución, transmitir todos los paquetes con valor $TS=n$ antes de transmitir un paquete con valor $TS=n+1$
- ◆ WFQ es costoso de ejecutar por lo que se han propuesto variantes que lo simplifican

Índice

- ♦ Introducción a la QoS
- ♦ Necesidades del tráfico de usuario
- ♦ Técnicas utilizadas en QoS
 - Control de admisión de conexiones
 - Encaminamiento con QoS
 - Conformado
 - Función Policía
 - Control de congestión
 - Control de flujo
 - Planificadores de tráfico
- ♦ Señalización

Señalización

- ◆ Necesaria para desplegar la QoS de manera escalable
- ◆ Objetivo: configurar cada nodo a lo largo de un camino, para que reconozca un determinado tráfico y le aplique un determinado tratamiento
- ◆ Ejemplos:
 - ATM: UNI, NNI, PNNI
 - IP: RSVP
- ◆ Cuando una red no tiene señalización, los nuevos servicios deben configurarse de forma manual
 - No escalable ni rentable
- ◆ NOTA: Aspectos de seguridad no tratados

Referencias

- ♦ J. Chao, X. Guo, "Quality of Service Control in High-Speed Networks" Editorial Wiley. "2002
- ♦ J. M. Arco, "Propuesta de optimización de la interconexión de redes con calidad de servicio para aplicaciones multimedia". Tesis Doctoral, Universidad de Alcalá, abril 2000. Disponible en it.aut.uah.es/josema