

# Programa de salida de QoS en los switches de la serie Catalyst 6500/6000 con software del sistema CatOS

## Contenido

[Introducción](#)

[Prerequisites](#)

[Requirements](#)

[Componentes Utilizados](#)

[Convenciones](#)

[Antecedentes](#)

[Perdidas de la cola de salida](#)

[Tipos de colas que participan en la programación de salida en Catalyst 6500/6000](#)

[Eliminación de cola](#)

[Detección temprana aleatoria y detección temprana aleatoria ponderada](#)

[Ordenamiento cíclico ponderado](#)

[Cola de prioridad estricta](#)

[Capacidad de colas de salida de diferentes tarjetas de línea en el Catalyst 6000](#)

[Capacidades del comando show port](#)

[Comprensión de la Capacidad de Colocación en Cola de un Puerto](#)

[Creación de QoS en Catalyst 6500/6000](#)

[Mecanismo de programación de salida en Catalyst 6500/6000](#)

[Configuración, Monitoreo y Programación de Salida en Catalyst 6500/6000](#)

[Configuración predeterminada para QoS en Catalyst 6500/6000](#)

[Configuración](#)

[Supervisión de la programación de resultados y verificación de la configuración](#)

[Utilice la programación de salida para reducir la demora y la fluctuación](#)

[Reducir el retraso](#)

[Reducción de la fluctuación](#)

[Información Relacionada](#)

## Introducción

La programación de salida se asegura de que el tráfico importante no se interrumpa en caso de que se produzca una suscripción excesiva. Este documento explica todas las técnicas y algoritmos implicados en la programación de salida en los Cisco Catalyst 6500/6000 Series Switches que ejecutan el software del sistema Catalyst OS (CatOS). Este documento también proporciona una breve descripción de la capacidad de puesta en cola de Catalyst 6500/6000 Switches y cómo configurar los diversos parámetros de la programación de salida.

**Nota:** Si ejecuta Cisco IOS® Software en su Catalyst 6500/6000, consulte [Programación de Salida QoS en Catalyst 6500/6000 Series Switches que Ejecutan Cisco IOS System Software](#) para obtener más información.

## Prerequisites

### Requirements

No hay requisitos específicos para este documento.

### Componentes Utilizados

Los ejemplos de este documento se crearon a partir de un Catalyst 6000 con un Supervisor Engine 1A y una Policy Feature Card (PFC). Pero los ejemplos también son válidos para un Supervisor Engine 2 con un PFC2 o para un Supervisor Engine 720 con un PFC3.

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, make sure that you understand the potential impact of any command.

### Convenciones

Consulte [Convenciones de Consejos Técnicos Cisco para obtener más información sobre las convenciones del documento.](#)

## Antecedentes

### Perdidas de la cola de salida

Las caídas de salida son causadas por una interfaz congestionada. Una causa común de esto podría ser el tráfico de un link de ancho de banda alto que se conmuta a un link de ancho de banda inferior o el tráfico de varios links entrantes que se conmutan a un único link de salida.

Por ejemplo, si una gran cantidad de tráfico en ráfaga entra en una interfaz gigabit y se conmuta hacia una interfaz de 100 Mbps, esto podría hacer que las caídas de salida aumenten en la interfaz de 100 Mbps. Esto ocurre porque la cola de salida en esa interfaz está saturada por el exceso de tráfico debido a la asimetría de la velocidad entre los anchos de banda entrante y saliente. La velocidad de tráfico en la interfaz saliente no puede aceptar todos los paquetes que se deben enviar.

La mejor solución para resolver el problema es aumentar la velocidad de la línea. Sin embargo, hay maneras de prevenir, disminuir o controlar las caídas de salida cuando no desea aumentar la velocidad de línea. Sólo puede evitar caídas de salida si las caídas de salida son consecuencia de ráfagas cortas de datos. Si las caídas de salida son causadas por un flujo constante de alta velocidad, no puede evitar las caídas. Sin embargo, puede controlarlos.

## Tipos de colas que participan en la programación de salida en Catalyst 6500/6000

## Eliminación de cola

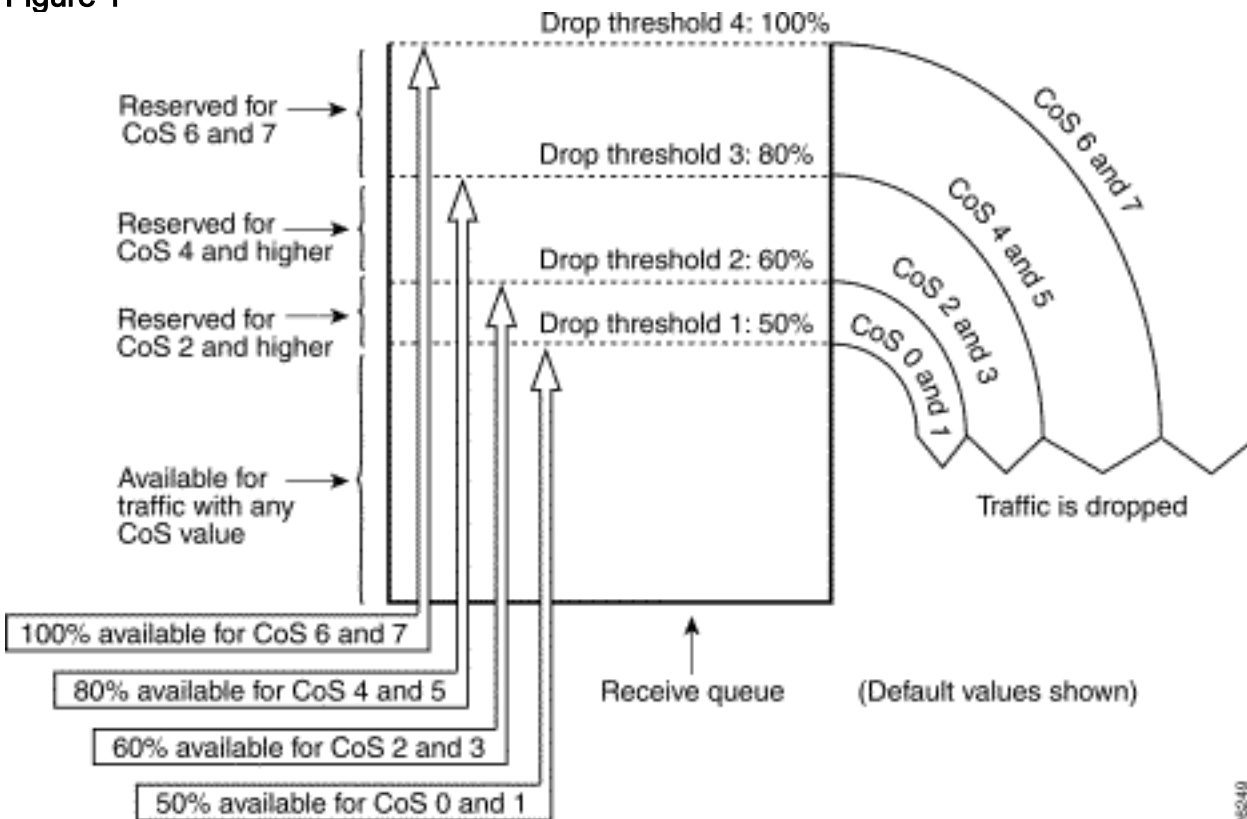
El descarte de cola es un mecanismo básico para evitar la congestión. El descarte de cola trata todo el tráfico de forma equitativa y no diferencia entre las clases de servicio (CoSs) cuando las colas empiezan a llenarse durante períodos de congestión. Cuando la cola de salida está llena y la caída de cola está en vigor, los paquetes se descartan hasta que se elimina la congestión y la cola ya no está llena. El descarte de cola es el tipo más básico de prevención de congestión y no tiene en cuenta ningún parámetro de QoS.

El Catalyst 6000 ha implementado una versión avanzada de la prevención de la congestión de caídas de cola que descarta todos los paquetes con un CoS determinado cuando se alcanza un determinado porcentaje de llenado de búfer. Con la eliminación de cola ponderada, puede definir un conjunto de umbrales y asociar una CoS a cada umbral. En el ejemplo de esta sección, hay cuatro umbrales posibles. Las definiciones de cada umbral son las siguientes:

- El umbral 1 se alcanza cuando se llena el 50% del búfer. Las CoS 0 y 1 se asignan a este umbral.
- El umbral 2 se alcanza cuando se llena el 60% del búfer. Las CoS 2 y 3 se asignan a este umbral.
- El umbral 3 se alcanza cuando se llena el 80% del búfer. Las CoS 4 y 5 se asignan a este umbral.
- El umbral 4 se alcanza cuando se llena el 100% del búfer. Las CoS 6 y 7 se asignan a este umbral.

En el diagrama de la [Figura 1](#), todos los paquetes con una CoS de 0 o 1 se descartan si el buffer está lleno al 50 por ciento. Todos los paquetes con una CoS de 0, 1, 2 o 3 se descartan si las memorias intermedias se llenan en un 60%. Los paquetes con una CoS de 6 ó 7 se descartan cuando las memorias intermedias están completas.

Figure 1



**Nota:** Tan pronto como el relleno del búfer cae por debajo de un cierto umbral, los paquetes con el CoS asociado ya no se descartan.

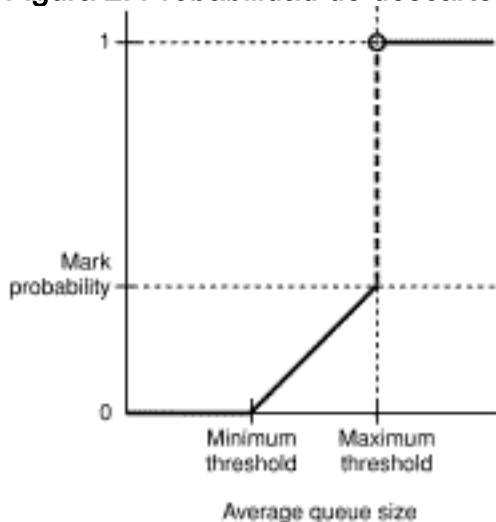
## Detección temprana aleatoria y detección temprana aleatoria ponderada

La detección temprana aleatoria ponderada (WRED) es un mecanismo de prevención de congestión que descarta aleatoriamente los paquetes con una determinada precedencia IP cuando las memorias intermedias alcanzan un umbral de llenado definido. WRED es una combinación de estas dos funciones:

- Eliminación de cola
- Detección temprana aleatoria (RED)

RED no reconoce la precedencia ni la CoS. RED utiliza uno de los umbrales únicos cuando se completa el valor de umbral para el búfer. RED comienza a descartar paquetes aleatoriamente (pero no todos los paquetes, como en la eliminación de cola) hasta que se alcanza el umbral máximo (máximo). Después de alcanzar el umbral máximo, se descartan todos los paquetes. La probabilidad de que se descarte un paquete aumenta linealmente con el aumento del relleno del búfer por encima del umbral. El diagrama de la [Figura 2](#) muestra la probabilidad de caída de paquetes:

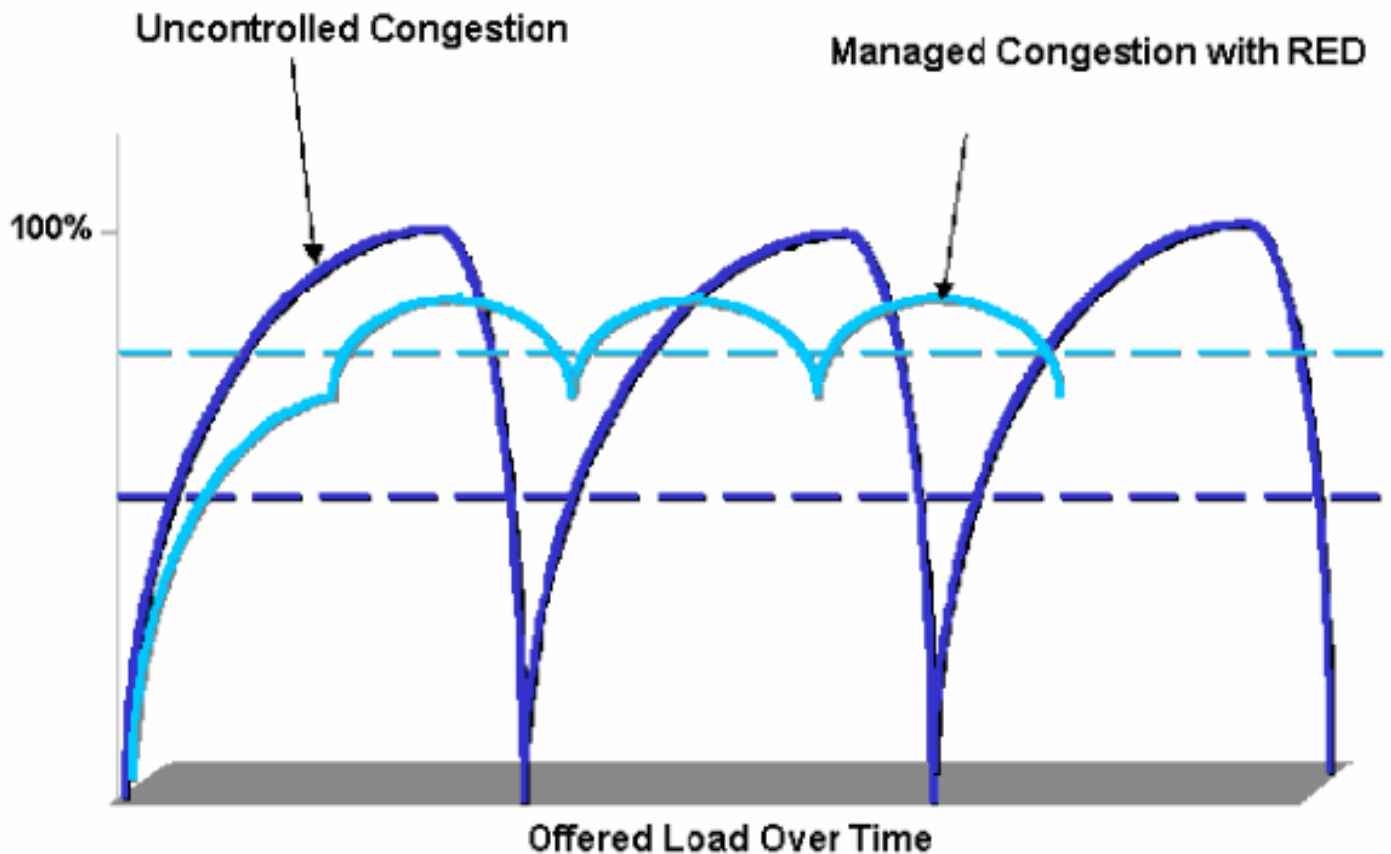
**Figura 2: Probabilidad de descarte de paquetes**



**Nota:** La probabilidad de marca en este diagrama es ajustable en RED, lo que significa que la pendiente de la probabilidad de caída lineal es ajustable.

RED y WRED son mecanismos muy útiles para evitar la congestión del tráfico basado en TCP. Para otros tipos de tráfico, RED no resulta muy eficiente. Esto se debe a que RED aprovecha el mecanismo de ventanas que TCP utiliza para administrar la congestión. RED evita la congestión típica que ocurre en un router cuando varias sesiones TCP pasan por el mismo puerto del router. El mecanismo se denomina sincronización de red global. El diagrama de la [Figura 3](#) muestra cómo el tipo de perfil ROJO produce un efecto suave en la carga:

**Figura 3: RED para evitar la congestión**



Para obtener más información sobre cómo RED puede reducir la congestión y suavizar el tráfico a través del router, refiérase a la sección [Cómo el Router Interactúa con TCP](#) del documento [Descripción General de la Prevención de Congestión](#).

WRED es similar a RED en que ambos definen algunos umbrales mínimos (min) y, cuando se alcanzan esos umbrales mínimos, los paquetes se descartan aleatoriamente. WRED también define ciertos umbrales máximos y, cuando se alcanzan esos umbrales máximos, se descartan todos los paquetes. WRED también reconoce CoS, lo que significa que se agregan uno o más valores CoS a cada par de umbral mínimo/máximo. Cuando se excede el umbral mínimo, los paquetes se descartan aleatoriamente con el CoS asignado. Considere este ejemplo con dos umbrales en la cola:

- Las CoS 0 y 1 se asignan al umbral mínimo 1 y al umbral máximo 1. El umbral mínimo 1 se establece en el 50% del relleno del búfer y el umbral máximo 1 se establece en el 80%.
- Las CoS 2 y 3 se asignan al umbral mínimo 2 y al umbral máximo 2. El umbral mínimo 2 se establece en el 70% del relleno del búfer y el umbral máximo 2 se establece en el 100%.

Tan pronto como el búfer excede el umbral mínimo 1 (50 por ciento), los paquetes con CoS 0 y 1 comienzan a descartarse aleatoriamente. Se descartan más paquetes a medida que aumenta la utilización del búfer. Si se alcanza el umbral mínimo 2 (70%), los paquetes con CoS 2 y 3 comienzan a descartarse aleatoriamente.

**Nota:** En esta etapa, la probabilidad de caída para los paquetes con CoS 0 y 1 es mucho mayor que la probabilidad de caída para los paquetes con CoS 2 o CoS 3.

Cada vez que se alcanza el umbral máximo 2, se descartan todos los paquetes con CoS 0 y 1, mientras que los paquetes con CoS 2 y 3 siguen descartándose aleatoriamente. Por último, cuando se alcanza el 100% (umbral máximo 2), se descartan todos los paquetes con CoS 2 y 3.

Los diagramas de la [Figura 4](#) y la [Figura 5](#) ilustran un ejemplo de estos umbrales:

Figura 4: WRED con dos conjuntos de umbrales mínimos y umbrales máximos (dos servicios)

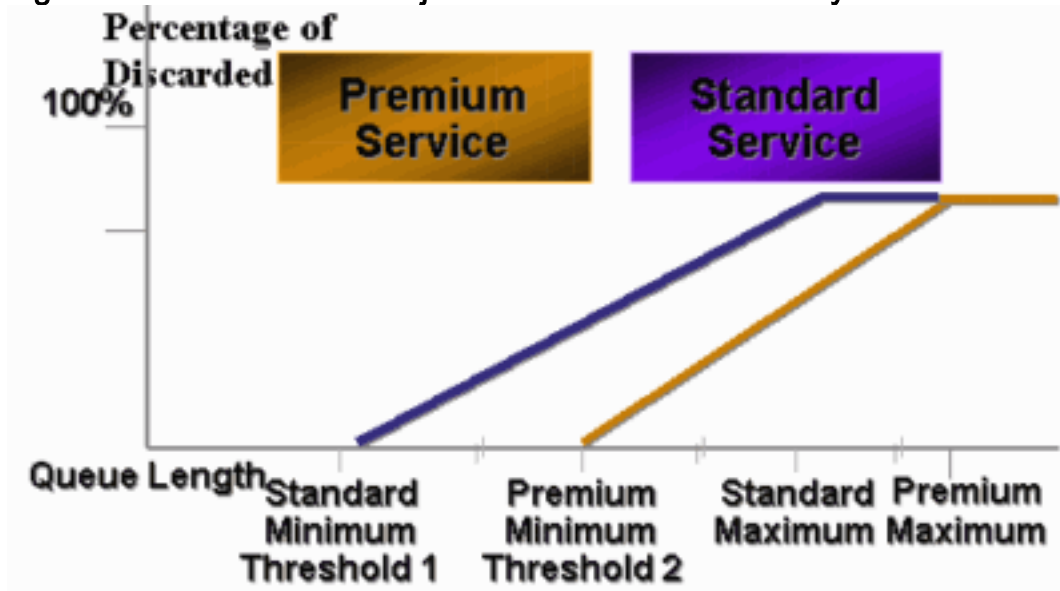
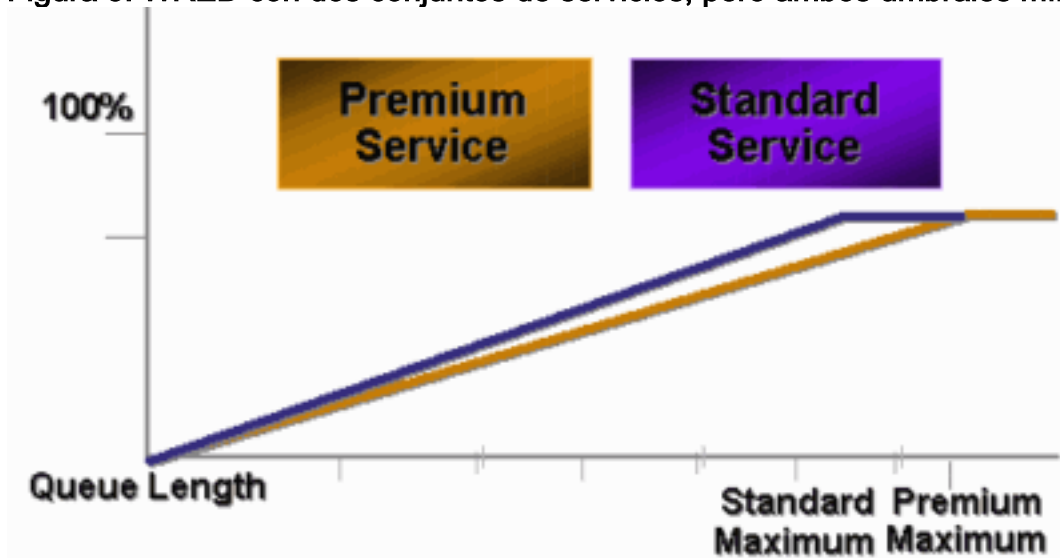


Figura 5: WRED con dos conjuntos de servicios, pero ambos umbrales mínimos son iguales a 0



La implementación temprana de CatOS de WRED sólo configuró el umbral máximo, mientras que el umbral mínimo estaba codificado en 0 por ciento. La parte inferior del diagrama de la [Figura 5](#) resalta el comportamiento resultante.

**Nota:** La probabilidad de caída para un paquete siempre es no nula porque esta probabilidad siempre está por encima del umbral mínimo. Este comportamiento se ha corregido en la versión de software 6.2 y posteriores.

### Ordenamiento cíclico ponderado

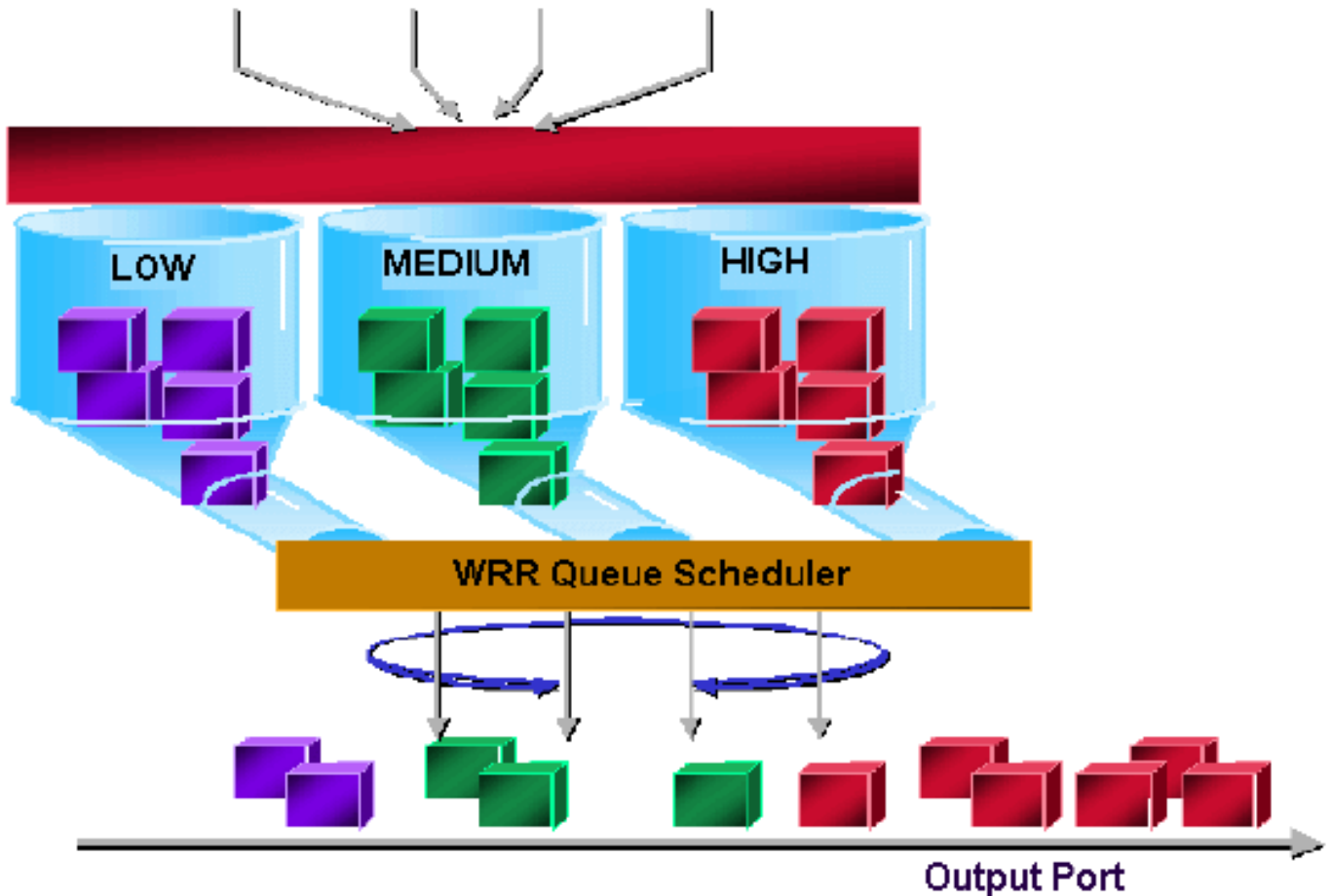
El ordenamiento cíclico ponderado (WRR) es otro mecanismo para la programación de salida en el Catalyst 6000. WRR funciona entre dos o más colas. Las colas para WRR se vacían de forma rotativa y puede configurar el peso para cada cola. De forma predeterminada, los puertos tienen dos colas WRR en el Catalyst 6000. El valor predeterminado es:

- Para prestar servicio a la cola WRR de alta prioridad el 70% del tiempo
- Para prestar servicio a la cola WRR de baja prioridad el 30% del tiempo

El diagrama de la [Figura 6](#) muestra un WRR que tiene tres colas que se atienden en un modo

WRR. La cola de alta prioridad (paquetes rojos) envía más paquetes que las otras dos colas:

Figura 6: Programación de salida: WRR



**Nota:** La mayoría de las tarjetas de línea 6500 implementan WRR por ancho de banda. Esta implementación de WRR por ancho de banda significa que cada vez que el planificador permite que una cola transmita paquetes, se permite transmitir un determinado número de bytes. Este número de bytes puede representar más de un paquete. Por ejemplo, si envía 5120 bytes en un turno, puede enviar tres paquetes de 1518 bytes, por un total de 4554 bytes. Los bytes en exceso se pierden ( $5120 - 4554 = 566$  bytes). Por lo tanto, con un peso extremo (como el 1% para la cola 1 y el 99% para la cola 2), es posible que no se alcance el peso configurado exacto. Esta incapacidad para alcanzar el peso exacto es a menudo el caso de los paquetes más grandes.

Algunas tarjetas de línea de nueva generación, como el 6548-RJ-45, superan esta limitación mediante la implementación de ordenamiento cíclico ponderado por déficit (DWRR). DWRR transmite desde las colas pero no deja pasar la cola de baja prioridad. DWRR realiza un seguimiento de la cola de baja prioridad que se está transmitiendo y compensa en la siguiente ronda.

### [Cola de prioridad estricta](#)

Otro tipo de cola en Catalyst 6000, una cola de prioridad estricta, siempre se vacía primero. En cuanto hay un paquete en la cola de prioridad estricta, se envía el paquete.

Las colas WRR o WRED se verifican sólo después de vaciar la cola de prioridad estricta. Después de que cada paquete se transmite desde la cola WRR o la cola WRED, la cola de prioridad estricta se verifica y se vacía, si es necesario.



**Nota:** Todas las tarjetas de línea con un tipo de cola similar a 1p2q1t, 1p3q8t y 1p7q8t utilizan DWRR. Otras tarjetas de línea utilizan WRR estándar.

## Capacidad de colas de salida de diferentes tarjetas de línea en el Catalyst 6000

### Capacidades del comando show port

Si no está seguro de la capacidad de colocación en cola de un puerto, puede ejecutar el comando **show port capabilities**. Este es el resultado del comando en una tarjeta de línea WS-X6408-GBIC:

```
Model                WS-X6408-GBIC
Port                 4/1
Type                 No GBIC
Speed                1000
Duplex                full
Trunk encap type     802.1Q,ISL
Trunk mode            on,off,desirable,auto,nonegotiate
Channel              yes
Broadcast suppression percentage(0-100)
Flow control          receive-(off,on,desired),send-(off,on,desired)
Security              yes
MembershIP            static,dynamic
Fast start            yes
QoS scheduling        rx-(1q4t),tx-(2q2t)
CoS rewrite           yes
ToS rewrite           DSCP
UDLD                  yes
SPAN                  source,destination
COPS port group       none
```

Este puerto tiene un tipo de salida de cola que se denomina 2q2t.

### Comprensión de la Capacidad de Colocación en Cola de un Puerto

Hay varios tipos de colas disponibles en los switches Catalyst 6500/6000. Las tablas de esta sección pueden quedar incompletas a medida que se liberen nuevas tarjetas de línea. Las nuevas tarjetas de línea pueden introducir nuevas combinaciones de colas. Para obtener una descripción actual de todas las colas disponibles para los módulos de switch Catalyst 6500/6000, refiérase a la sección *Configuración de QoS* para su versión CatOS de la [Documentación de Software Catalyst 6500 Series](#).

**Nota:** Cisco Communication Media Module (CMM) no admite todas las funciones de QoS. Verifique las notas de la versión para su versión de software específica para determinar las funciones soportadas.

Esta tabla explica la notación de la arquitectura de QoS del puerto:

ide Tx <sup>1</sup> / Rx <sup>2</sup>	Notación de cola	Nº de colas	cola prioritaria	Nº de colas WRR	Nº y tipo de umbral para colas WRR
Tr	2q2t	2	—	2	2 caídas de cola



					configurables
Tr	1p2q2t	3	1	2	2 WRED configurables
Tr	1p3q1t	4	1	3	1 WRED configurable
Tr	1p2q1t	3	1	2	1 WRED configurable
Rx	1q4t	1	—	1	4 caídas de cola configurables
Rx	1p1q4t	2	1	1	4 caídas de cola configurables
Rx	1p1q0t	2	1	1	No configurable
Rx	1p1q8t	2	1	1	8 WRED configurables

<sup>1</sup> Tx = transmisión.

<sup>2</sup> Rx = recibir.

Esta tabla enumera todos los módulos y los tipos de cola en el lado Rx y Tx de la interfaz o puerto:

Módulo	Colas Rx	Colas Tx
WS-X6K-S2-PFC2	1p1q4t	1p2q2t
WS-X6K-SUP1A-2GE	1p1q4t	1p2q2t
WS-X6K-SUP1-2GE	1q4t	2q2t
WS-X6501-10GEX4	1p1q8t	1p2q1t
WS-X6502-10GE	1p1q8t	1p2q1t
WS-X6516-GBIC	1p1q4t	1p2q2t
WS-X6516-GE-TX	1p1q4t	1p2q2t
WS-X6416-GBIC	1p1q4t	1p2q2t
WS-X6416-GE-MT	1p1q4t	1p2q2t
WS-X6316-GE-TX	1p1q4t	1p2q2t
WS-X6408A-GBIC	1p1q4t	1p2q2t
WS-X6408-GBIC	1q4t	2q2t
WS-X6524-100FX-MM	1p1q0t	1p3q1t
WS-X6324-100FX-SM	1q4t	2q2t
WS-X6324-100FX-MM	1q4t	2q2t
WS-X6224-100FX-MT	1q4t	2q2t
WS-X6548-RJ-21	1p1q0t	1p3q1t
WS-X6548-RJ-45	1p1q0t	1p3q1t
WS-X6348-RJ-21	1q4t	2q2t
WS-X6348-RJ21V	1q4t	2q2t

WS-X6348-RJ-45	1q4t	2q2t
WS-X6348-RJ-45V	1q4t	2q2t
WS-X6148-RJ-45V	1q4t	2q2t
WS-X6148-RJ21V	1q4t	2q2t
WS-X6248-RJ-45	1q4t	2q2t
WS-X6248A-TEL	1q4t	2q2t
WS-X6248-TEL	1q4t	2q2t
WS-X6024-10FL-MT	1q4t	2q2t

## [Creación de QoS en Catalyst 6500/6000](#)

Se utilizan tres campos en el Catalyst 6500/6000 para realizar QoS:

- Precedencia IP: los primeros tres bits del campo Tipo de servicio (ToS) en el encabezado IP
- Punto de código de servicios diferenciados (DSCP): los primeros seis bits del campo ToS en el encabezado IP
- CoS: los tres bits utilizados en el nivel de capa 2 (L2) Estos tres bits forman parte del encabezado Inter-Switch Link (ISL) o están dentro de la etiqueta IEEE 802.1Q (dot1q). No hay CoS dentro de un paquete Ethernet sin etiqueta.

## [Mecanismo de programación de salida en Catalyst 6500/6000](#)

Cuando se envía una trama desde el bus de datos para ser transmitida, el CoS del paquete es el único parámetro que se considera. Luego, el paquete pasa a través de un planificador, que elige la cola en la que se coloca el paquete. Por lo tanto, recuerde que la programación de salida y todos los mecanismos que se describen en este documento sólo reconocen CoS.

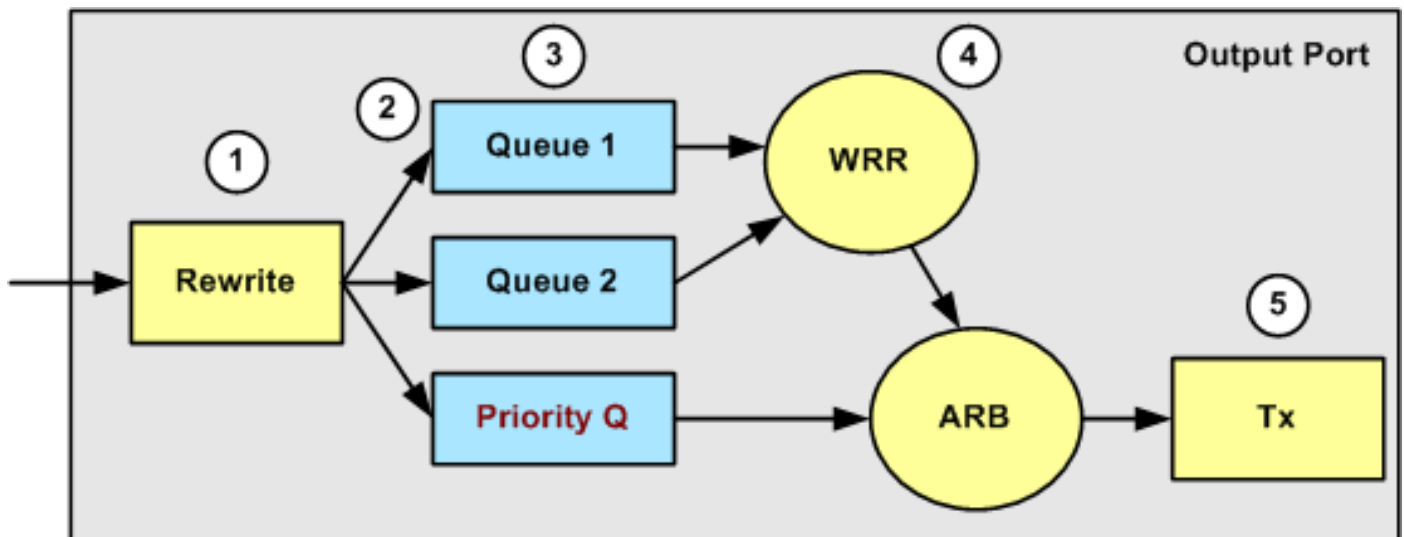
El Catalyst 6500/6000 con una tarjeta de función de switch multicapa (MSFC) utiliza un DSCP interno para clasificar el paquete. El Catalyst 6500/6000 configurado con QoS habilitado asigna un valor DSCP cuando se toma la decisión de reenvío en el nivel PFC. Este DSCP se asigna a cualquier paquete, que incluye paquetes que no son de IP, y se asigna al CoS para habilitar la programación de salida. Puede configurar el mapping de valores DSCP a CoS en el Catalyst 6500/6000. Si deja el valor predeterminado, puede derivar el CoS del DSCP. La fórmula es:

$$DSCP\_value / 8$$

Además, el valor DSCP se mapea en el CoS del paquete saliente, si el paquete es un paquete IP etiquetado ISL o dot1q (VLAN no nativa). El valor DSCP también se escribe dentro del campo ToS del encabezado IP.

El diagrama de la [Figura 7](#) muestra una cola 1p2q2t. Las colas WRR se vacían con el uso del planificador WRR. También hay un árbitro que verifica entre cada paquete de las colas WRR para determinar si hay algo en la cola de prioridad estricta.

### Figura 7



1. El campo ToS se reescribe en el encabezado IP y en el campo 802.1p/ISL CoS.
2. La cola de programación y el umbral se seleccionan en función del CoS, a través de un mapa configurable.
3. Cada cola tiene un tamaño y umbrales configurables, y algunas colas tienen WRED.
4. La eliminación de cola utiliza WRR entre dos colas.
5. La encapsulación saliente puede ser dot1q, ISL o none.

## [Configuración, Monitoreo y Programación de Salida en Catalyst 6500/6000](#)

### [Configuración predeterminada para QoS en Catalyst 6500/6000](#)

Esta sección proporciona un ejemplo de salida de la configuración de QoS predeterminada en un Catalyst 6500/6000, además de información sobre lo que estos valores significan y cómo puede ajustar los valores.

QoS se inhabilita de forma predeterminada cuando se ejecuta este comando:

```
set qos disable
```

Los comandos de esta lista muestran la asignación predeterminada para cada CoS en un puerto 2q2t. La cola 1 tiene CoS 0 y 1 asignado a su primer umbral y tiene CoS 2 y 3 asignado a su segundo umbral. La cola 2 tiene CoS 4 y 5 asignados a su primer umbral y tiene CoS 6 y 7 asignados a su segundo umbral:

```
set qos map 2q2t tx 1 1 cos 0
```

```
set qos map 2q2t tx 1 1 cos 1
```

```
set qos map 2q2t tx 1 2 cos 2
```

```
set qos map 2q2t tx 1 2 cos 3
```

```
set qos map 2q2t tx 2 1 cos 4
```

```
set qos map 2q2t tx 2 1 cos 5
```

```
set qos map 2q2t tx 2 2 cos 6
```

```
set qos map 2q2t tx 2 2 cos 7
```

Estos comandos muestran el nivel de umbral de forma predeterminada en un puerto 2q2t para cada cola:

```
set qos drop-threshold 2q2t tx queue 1 80 100
```

```
set qos drop-threshold 2q2t tx queue 2 80 100
```

Puede asignar el peso predeterminado a cada una de las colas WRR. Ejecute este comando para asignar los pesos predeterminados para la cola 1 y la cola 2:

**Nota:** La cola de baja prioridad se atiende el 5/260% del tiempo y la cola de alta prioridad se atiende el 255/260% del tiempo.

```
set qos wrr 2q2t 5 255
```

La disponibilidad total del búfer se divide entre las dos colas. La cola de baja prioridad se asigna correctamente al 80% de los búfers disponibles porque ésta es la cola que probablemente tenga paquetes almacenados en búfer y sentados durante algún tiempo. Ejecute este comando para definir la disponibilidad:

```
set qos txq-ratio 2q2t 80 20
```

Puede ver configuraciones similares para el puerto 1p2q2t en esta configuración:

```
set qos map 1p2q2t tx 1 1 cos 0
```

```
set qos map 1p2q2t tx 1 1 cos 1
```

```
set qos map 1p2q2t tx 1 2 cos 2
```

```
set qos map 1p2q2t tx 1 2 cos 3
```

```
set qos map 1p2q2t tx 2 1 cos 4
```

```
set qos map 1p2q2t tx 3 1 cos 5
```

```
set qos map 1p2q2t tx 2 1 cos 6
```

```
set qos map 1p2q2t tx 2 2 cos 7
```

```
set qos wrr 1p2q2t 5 255
```

```
set qos txq-ratio 1p2q2t 70 15 15
set qos wred 1p2q2t tx queue 1 80 100
set qos wred 1p2q2t tx queue 2 80 100
```

**Nota:** De forma predeterminada, el CoS 5 (tráfico de voz) se asigna a la cola de prioridad estricta.

## Configuración

El primer paso de la configuración es habilitar QoS. Recuerde que QoS está desactivada de manera predeterminada. Cuando se inhabilita QoS, la asignación de CoS es irrelevante. Hay una cola única que se sirve como FIFO y todos los paquetes se descartan allí.

```
bratan> (enable) set qos enable
```

```
QoS is enabled
```

```
bratan> (enable) show qos status
```

```
QoS is enabled on this switch
```

El valor CoS debe asignarse a la cola o al umbral para todos los tipos de cola. El mapping definido para un tipo de puerto 2q2t no se aplica a ningún puerto 1p2q2t. Además, el mapping que se hace para 2q2t se aplica a todos los puertos que tienen un mecanismo de colocación en cola 2q2t. Ejecutar este comando:

```
set qos map queue_type tx Q_number threshold_number cos value
```

**Nota:** Las colas siempre están numeradas para comenzar con la cola de prioridad más baja posible y terminar con la cola de prioridad estricta disponible. Aquí tiene un ejemplo:

- La cola 1 es la cola WRR de baja prioridad
- La cola 2 es la cola WRR de alta prioridad
- La cola 3 es la cola de prioridad estricta

Debe repetir esta operación para todos los tipos de colas. De lo contrario, mantenga la asignación de CoS predeterminada. Este es un ejemplo para 1p2q2t:

## Configuración

```
set qos map 1p2q2t tx 1 1 cos 0
!--- This is the low-priority WRR queue threshold 1, CoS 0 and 1. set qos map 1p2q2t tx 1 1 cos
1 and 1
```

```
set qos map 1p2q2t tx 1 2 cos 2
!--- This is the low-priority WRR queue threshold 2, CoS 2 and 3. set qos map 1p2q2t tx 1 2 cos
3 and 3
```

```
set qos map 1p2q2t tx 2 1 cos 4
!--- This is the high-priority WRR queue threshold 1, CoS 4. set qos map 1p2q2t tx 3 1 cos 5
!--- This is the strict priority queue, CoS 5. set qos map 1p2q2t tx 2 1 cos 6
```

```
!--- This is the high-priority WRR queue threshold 2, CoS 6. set qos map 1p2q2t tx 2 2 cos 7 and 7
```

## [Resultado de la consola](#)

```
tamer (enable) set qos map 1p2q2t tx 1 1 cos 0
```

QoS tx priority queue and threshold mapped to cos successfully

Debe configurar el peso WRR para las dos colas WRR. Ejecutar este comando:

```
set qos wrr Q_type weight_1 weight_2
```

*Weight\_1* se relaciona con la cola 1, que debe ser la cola WRR de baja prioridad. *Weight\_1* siempre debe ser menor que *weight\_2*. El peso puede tomar cualquier valor entre 1 y 255. Puede asignar el porcentaje con estas fórmulas:

- Cola 1:

$$\text{weight}_1 / (\text{weight}_1 + \text{weight}_2)$$

- Cola 2:

$$\text{weight}_2 / (\text{weight}_1 + \text{weight}_2)$$

También debe definir el peso para los distintos tipos de colas. El peso no necesita ser el mismo. Por ejemplo, para 2q2t, donde la cola 1 se atiende el 30% del tiempo y la cola 2 se atiende el 70% del tiempo, puede ejecutar este comando para definir el peso:

```
set qos wrr 2q2t 30 70
```

```
!--- This ensures that the high-priority WRR queue is served 70 percent of the time !--- and that the low-priority WRR queue is served 30 percent of the time.
```

## [Resultado de la consola](#)

```
tamer (enable) set qos wrr 2q2t 30 70
```

QoS wrr ratio is set successfully

También debe definir la relación de cola de transmisión, que se refiere a la forma en que los búfers se dividen entre las diferentes colas. Ejecutar este comando:

```
set qos txq-ratio port_type queue1_val queue2_val ... queueN_val
```

**Nota:** Si tiene tres colas (1p2q2t), debe establecer la cola WRR de alta prioridad y la cola de prioridad estricta en el mismo nivel por razones de hardware.

## [Configuración](#)

```
set qos txq-ratio 1p2q2t 70 15 15
```

*!--- This gives 70 percent of the buffer of all 1p2q2t ports to the low-priority WRR !--- queue and gives 15 percent to each of the other two queues. set qos txq-ratio 2q2t 80 20 !--- This gives 80 percent of the buffer to the low-priority queue, !--- and gives 20 percent of the buffer to the high-priority queue.*

## Resultado de la consola

```
tamer (enable) set qos txq-ratio 1p2q2t 70 15 20
```

Queue ratio values must be in range of 1-99 and add up to 100  
Example: set qos txq-ratio 2q2t 20 80

```
tamer (enable) set qos txq-ratio 1p2q2t 70 30 30
```

Queue ratio values must be in range of 1-99 and add up to 100  
Example: set qos txq-ratio 2q2t 20 80

```
tamer (enable) set qos txq-ratio 1p2q2t 80 10 10
```

QoS txq-ratio is set successfully

Como ilustra este resultado de la consola, la suma de los valores de cola debe ser 100. Deje la mayor parte de las memorias intermedias para la cola WRR de baja prioridad porque esta cola necesita el mayor almacenamiento en búfer. Las otras colas tienen prioridad de servicio más alta.

El último paso es configurar el nivel del umbral para la cola WRED o para la cola de eliminación. Ejecute estos comandos:

```
set qos wred port_type [tx] queue q_num thr1 thr2 ... thrn
```

```
set qos drop-threshold port_type tx queue q_num thr1 ... thr2
```

## Configuración

```
set qos drop-threshold 2q2t tx queue 1 50 80
```

*!--- For low-priority queues in the 2q2t port, the first threshold is defined at 50 !--- percent and the second threshold is defined at 80 percent of buffer filling. set qos drop-threshold 2q2t tx queue 2 40 80*

*!--- For high-priority queues in the 2q2t port, the first threshold is defined at 40 !--- percent and the second threshold is defined at 80 percent of buffer filling. set qos wred 1p2q2t tx queue 1 50 90*

*!--- The commands for the 1p2q2t port are identical. set qos wred 1p2q2t tx queue 2 40 80*

## Resultado de la consola

```
tamer (enable) set qos drop-threshold 2q2t tx queue 1 50 80
```

Transmit drop thresholds for queue 1 set at 50% 80%

```
tamer (enable) set qos drop-threshold 2q2t tx queue 2 40 80
```

Transmit drop thresholds for queue 2 set at 40% 80%



```
tamer (enable) set qos wred 1p2q2t tx queue 1 50 90
```

WRED thresholds for queue 1 set to 50 and 90 on all WRED-capable 1p2q2t ports

```
tamer (enable) set qos wred 1p2q2t tx queue 2 40 80
```

WRED thresholds for queue 2 set to 40 and 80 on all WRED-capable 1p2q2t ports

El comando **set qos wred 1p2q2t tx queue 2 40 80** funciona junto con el CoS para la asignación de umbral. Por ejemplo, cuando ejecuta los comandos en la siguiente lista, se asegura de que, en el puerto 1p2q2t en la dirección de transmisión, los paquetes con CoS 0,1, 2 y 3 se envíen en la primera cola (la cola WRR baja). Cuando las memorias intermedias en esa cola están llenas al 50%, WRED comienza a descartar paquetes con CoS 0 y 1. Los paquetes con CoS 2 y 3 se descartan sólo cuando las memorias intermedias de la cola están 90 por ciento rellenas.

```
set qos map 1p2q2t tx 1 1 cos 0
```

```
set qos map 1p2q2t tx 1 1 cos 1
```

```
set qos map 1p2q2t tx 1 2 cos 2
```

```
set qos map 1p2q2t tx 1 2 cos 3
```

```
set qos wred 1p2q2t tx queue 1 50 90
```

## [Supervisión de la programación de resultados y verificación de la configuración](#)

Un comando simple que se debe utilizar para verificar la configuración actual del tiempo de ejecución para la programación de salida de un puerto es **show qos info runtime mod/port**. El comando muestra esta información:

- El tipo de cola en el puerto
- Asignación de CoS a las diferentes colas y umbrales
- Uso compartido de búfer
- El peso WRR

En este ejemplo, los valores están en el 20% de WRR para la cola 1 y en el 80% de WRR para la cola 2:

```
tamer (enable) show qos info runtime 1/1
```

```
Run time setting of QoS:
```

```
QoS is enabled
```

```
Policy Source of port 1/1: Local
```

```
Tx port type of port 1/1 : 1p2q2t
```

```
Rx port type of port 1/1 : 1p1q4t
```

```
Interface type: port-based
```

```
ACL attached:
```

```
The qos trust type is set to untrusted
```

```
Default CoS = 0
```

```
Queue and Threshold Mapping for 1p2q2t (tx):
```

Queue	Threshold	CoS
1	1	0 1
1	2	2 3
2	1	4 6

```

2          2          7
3          1          5

```

Queue and Threshold Mapping for lplq4t (rx):

All packets are mapped to a single queue

Rx drop thresholds:

Rx drop thresholds are disabled

Tx drop thresholds:

Tx drop-thresholds feature is not supported for this port type

Tx WRED thresholds:

Queue #            Thresholds - percentage (\* abs values)

```

-----
1          80% (249088 bytes) 100% (311168 bytes)
2          80% (52480 bytes) 100% (61440 bytes)

```

Queue Sizes:

Queue #            Sizes - percentage (\* abs values)

```

-----
1          70% (311296 bytes)
2          15% (65536 bytes)
3          15% (65536 bytes)

```

WRR Configuration of ports with speed 1000Mbps:

Queue #            Ratios (\* abs values)

```

-----
1          20 (5120 bytes)
2          80 (20480 bytes)

```

(\*) Runtime information may differ from user configured setting due to hardware granularity.

tamer (enable)

En el siguiente ejemplo, observe que los pesos WRR no son el valor predeterminado de 1. Los pesos se han establecido en los valores 20 para la cola 1 y 80 para la cola 2. Este ejemplo utiliza un generador de tráfico para enviar 2 Gb de tráfico a un Catalyst 6000. Estos 2 Gb de tráfico deben salir por el puerto 1/1. Debido a que el puerto 1/1 está sobresuscrito, muchos paquetes se descartan (1 Gbps). El comando **show mac** muestra que hay un montón de caídas de salida:

tamer (enable) **show mac 1/1**

```

Port          Rcv-Unicast          Rcv-Multicast          Rcv-Broadcast
-----
1/1           0                    1239                   0

Port          Xmit-Unicast          Xmit-Multicast          Xmit-Broadcast
-----
1/1          73193601              421                    0

Port          Rcv-Octet            Xmit-Octet
-----
1/1          761993                100650803690

MAC          Dely-Exced            MTU-Exced            In-Discard            Out-Discard
-----
1/1           0                    -                    0                    120065264

```

Last-Time-Cleared

-----

Fri Jan 12 2001, 17:37:43

Tenga en cuenta los paquetes suprimidos. Así es como se divide el patrón de tráfico sugerido:

- 1 Gb de tráfico con precedencia IP 0
- 250 Mb de tráfico con precedencia IP 4
- 250 Mb de tráfico con precedencia IP 5

- 250 Mb de tráfico con precedencia IP 6
- 250 Mb de tráfico con precedencia IP 7

Según la asignación de CoS, este tráfico se envía:

- 1 Gb de tráfico a cola 1 umbral 1
- 0 Mb de tráfico a cola 1 umbral 2
- 500 Mb de tráfico a cola 2 umbral 1
- 250 Mb de tráfico a cola 2 umbral 2
- 250 Mb de tráfico a cola 3 (cola de prioridad estricta)

El switch debe confiar en el tráfico recibido para que la precedencia IP entrante se conserve en el switch y se utilice para asignar al valor CoS para la programación de salida.

**Nota:** La precedencia de IP predeterminada para la asignación de CoS es precedencia de IP igual a CoS.

Ejecute el comando **show qos stat 1/1** para ver los paquetes que se descartaron y el porcentaje aproximado:

- En este momento, no se descartan paquetes en la cola 3 (CoS 5).
- El 91,85% de los paquetes descartados son paquetes CoS 0 en la cola 1.
- El 8% de los paquetes perdidos son CoS 4 y 6 en la cola 2, umbral 1.
- 0.15 por ciento de los paquetes perdidos son CoS 7 en la cola 2, umbral 2.

Este resultado ilustra el uso del comando:

```
tamer (enable) show qos stat 1/1
```

```
Tx port type of port 1/1 : lp2q2t
Q3T1 statistics are covered by Q2T2.
Q #      Threshold #:Packets dropped
-----
1        1:110249298 pkts, 2:0 pkts
2        1:9752805 pkts, 2:297134 pkts
3        1:0 pkts
Rx port type of port 1/1 : lp1q4t
Rx drop threshold counters are disabled for untrusted ports
Q #      Threshold #:Packets dropped
-----
1        1:0 pkts, 2:0 pkts, 3:0 pkts, 4:0 pkts
2        1:0 pkts
```

Si vuelve a cambiar el peso de WRR al valor predeterminado después de que se hayan borrado los contadores, sólo el 1% de los paquetes perdidos se produce en la cola 2 en lugar del 8% que apareció anteriormente:

**Nota:** El valor predeterminado es 5 para la cola 1 y 255 para la cola 2.

```
tamer (enable) show qos stat 1/1
```

```
TX port type of port 1/1 : lp2q2t
Q3T1 statistics are covered by Q2T2
Q #      Threshold #:Packets dropped
-----
1        1:2733942 pkts, 2:0 pkts
2        1:28890 pkts, 2:6503 pkts
```

```

3          1:0 pkts
Rx port type of port 1/1 : lp1q4t
Rx drop threshold counters are disabled for untrusted ports
Q #          Threshold #:Packets dropped
---          -----
1          1:0 pkts, 2:0 pkts, 3:0 pkts, 4:0 pkts
2          1:0 pkts

```

## [Utilice la programación de salida para reducir la demora y la fluctuación](#)

En el ejemplo de la sección [Supervisión de la Programación de Resultados y Verificación de la Configuración](#) se muestra la ventaja de la implementación de la programación de salida, que evita una caída del tráfico VoIP o de misión crítica en caso de una suscripción excesiva del puerto de salida. La suscripción excesiva se produce con poca frecuencia en una red normal, especialmente en un link Gigabit. Por lo general, la suscripción excesiva solo se produce durante las horas pico de tráfico o durante las ráfagas de tráfico en un período de tiempo muy corto.

Incluso sin ninguna suscripción excesiva, la programación de salida puede ser de gran ayuda en una red donde la QoS se implementa de extremo a extremo. La programación de salida ayuda a reducir el retraso y la fluctuación. Esta sección proporciona ejemplos de cómo la programación de resultados puede ayudar a reducir la demora y la fluctuación.

### [Reducir el retraso](#)

El retraso de un paquete se incrementa por el tiempo "perdido" en el buffer de cada switch durante la espera de transmisión. Por ejemplo, un paquete de voz pequeño con un CoS de 5 se envía desde un puerto durante una copia de seguridad o transferencia de archivos de gran tamaño. Si no tiene QoS para el puerto de salida, y si asume que el paquete de voz pequeño está en cola después de 10 paquetes grandes de 1500 bytes, puede calcular fácilmente el tiempo de velocidad Gigabit necesario para transmitir los 10 paquetes grandes:

`(10 × 1500 × 8) = 120,000 bits that are transmitted in 120 microseconds`

Si este paquete necesita atravesar ocho o nueve switches mientras pasa a través de la red, puede producirse un retraso de aproximadamente 1 ms. Esta cantidad cuenta solamente los retrasos en la cola de salida del switch que se cruza en la red.

**Nota:** Si necesita poner en cola los mismos 10 paquetes grandes en una interfaz de 10 Mbps (por ejemplo, con un teléfono IP y con un PC conectado), la demora que se introduce es:

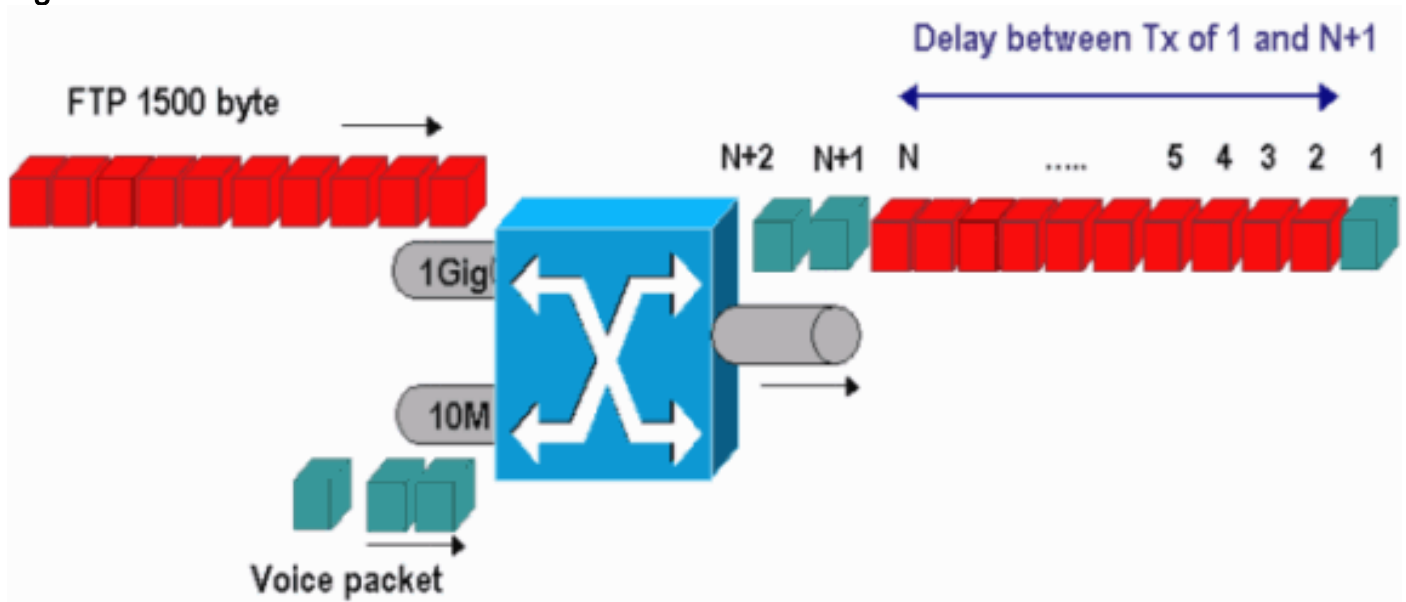
`(10 × 1500 × 8) = 120,000 bits that are transmitted in 12 ms`

La implementación de la programación de salida garantiza que los paquetes de voz con una CoS de 5 se coloquen en la cola de prioridad estricta. Esta colocación garantiza que estos paquetes se envíen antes que cualquier paquete con una CoS inferior a 5, lo que reduce los retrasos.

### [Reducción de la fluctuación](#)

Otra ventaja importante de la implementación de la programación de salida es que reduce la fluctuación. La fluctuación es la variación en la demora que se observa para los paquetes dentro del mismo flujo. El diagrama de la [Figura 8](#) muestra un escenario de ejemplo de cómo la programación de salida puede reducir la fluctuación:

Figura 8



En este escenario, hay dos secuencias que un único puerto de salida debe enviar:

- Un flujo de voz entrante en un puerto Ethernet de 10 Mbps
- Una secuencia FTP que se recibe en un enlace ascendente Ethernet de 1 Gbps

Ambos flujos dejan el switch a través del mismo puerto de salida. Este ejemplo muestra lo que puede suceder sin el uso de la programación de salida. Todos los paquetes de datos grandes pueden entrelazarse entre dos paquetes de voz, lo que crea fluctuación en la recepción del paquete de voz desde el mismo flujo. Hay una demora mayor entre la recepción del paquete  $n$  y el paquete  $n+1$  a medida que el switch transmite el paquete de datos grande. Sin embargo, el retraso entre  $n+1$  y  $n+2$  es insignificante. Esto produce fluctuación en el flujo de tráfico de voz. Puede evitar fácilmente este problema con el uso de una cola de prioridad estricta. Asegúrese de que el valor CoS de los paquetes de voz esté asignado a la cola de prioridad estricta.

## [Información Relacionada](#)

- [Programación de salida de QoS en switches Catalyst 6500/6000 Series que ejecutan Cisco IOS System Software](#)
- [La calidad del servicio en la familia de switches Catalyst 6000](#)
- [Páginas de Soporte de Productos de LAN](#)
- [Página de Soporte de LAN Switching](#)
- [Soporte Técnico y Documentación - Cisco Systems](#)