

Guía de solución de problemas de QoS para NCS serie 5500

Contenido

[Introducción](#)

[Implementación de QoS en NCS5500](#)

[Pautas para la resolución de problemas](#)

[QoS de ingreso](#)

[QoS de salida](#)

[Supervisar los FMQ](#)

[Comandos adicionales a recopilar](#)

[Limitaciones/advertencias](#)

[Marcación](#)

[Ancho de banda](#)

[QoS de ingreso](#)

[QoS de salida](#)

[HQoS \(calidad de servicio jerárquica\)](#)

[Advertencia de configuración no admitida después de la confirmación](#)

Introducción

Este documento describe la guía de solución de problemas de QoS para Cisco Network Convergence System (NCS) serie 5500.

Implementación de QoS en NCS5500

- La QoS en NCS5500 se realiza en la TM de ingreso y egreso de la NPU. NCS5500 implementa una arquitectura de cola de salida virtual, donde la cola de salida se basa en solicitudes de crédito y esquemas de concesión.
- El almacenamiento en búfer real ocurre en los dispositivos de reenvío de ingreso. El dispositivo de reenvío de egreso decide cuánta cantidad de tráfico se puede enviar concediendo créditos a cualquier dispositivo de reenvío de ingreso, que solicita crédito.
- Todas las colas de cualquier interfaz de salida o elemento de programación se replican en todos los dispositivos de ingreso que deseen enviar paquetes a esa interfaz. Estas colas se denominan Colas de salida virtuales (VOQ).
- Efectivamente, se admiten 8 colas por interfaz o elemento de programación. Al salir, el elemento de programación admite prioridad, min, max y exceso. La programación proporciona créditos en función de estos atributos.
- Por consiguiente, los atributos de programación de una interfaz se configuran en la salida,

mientras que los valores de control de congestión y prevención de congestión se configuran en la entrada.

- Cuando el IRPP decide enviar un paquete para un puerto de destino/salida dado, coloca el paquete en un VOQ específico. Durante la colocación en cola del paquete, se verifica si el paquete se puede poner en cola basándose en él.
- En la configuración de ocupación de búfer de VOQ y cola de estadísticas virtuales (VSQ) de la interfaz de ingreso. VSQ es un esquema de administración de búfer basado en clases de tráfico.
- Si se exceden las memorias intermedias, el paquete se descarta o el control de flujo se envía al puerto de ingreso según la configuración de usuario correspondiente. De forma predeterminada, los paquetes se descartan si fallan la verificación de la cola.
- Las LC del NCS5500 no tienen ASIC de colocación en cola independientes. La plataforma implementa la colocación en cola virtual basada en resultados, que se basa en la programación. Por lo tanto, no hay concepto de colocación en cola de ingreso. Políticas de servicio de ingreso que admiten acciones de marcado así como regulación únicamente.
- NCS5500 implementa un concepto diferente para el marcado y la colocación en cola. Se basa en qos-group y traffic-class. En la salida, estos dos parámetros se corresponden para marcar el tráfico y seleccionar colas.

Limitaciones existentes para varios comandos. Para obtener más información, *consulte* el .

Pautas para la resolución de problemas

QoS de ingreso

Uno de los comandos más útiles e importantes es **show qos interface <interface> input|output**

Proporciona información útil sobre la política de servicio aplicada. Muestra los parámetros configurados y convertidos (en el caso del regulador, modelador, umbrales de descarte de cola, etc.), los ID de cola virtual, el tipo de contabilidad, etc.

Comience con una política simple de ingreso.

```
class-map match-any i_match
match precedence ipv4 3
match mpls experimental topmost 3
end-class-map
!

policy-map ingress_marking
class i_match
set qos-group 3
set discard-class 2
!
class class-default
!
```

end-policy-map
!

```
RP/0/RP0/CPU0:xrg-ncs-04#show qos interface hundredGigE 0/0/0/35 input
```

Mon Dec 4 21:30:44.131 UTC

NOTE:- Configured values are displayed within parentheses

Interface HundredGigE0/0/0/35 ifh 0x138 -- input policy

NPU Id: 5

Total number of classes: 2

Interface Bandwidth: 100000000 kbps

Policy Name: **ingress_marking**

Accounting Type: Layer1 (Include Layer 1 encapsulation and above)

Level1 Class = **i_match**

New qos group = 3

New discard class = 2

Default Policer Bucket ID = 0x21

Default Policer Stats Handle = 0x0

Policer not configured for this class

Level1 Class = **class-default**

Default Policer Bucket ID = 0x20

Default Policer Stats Handle = 0x0

Policer not configured for this class

El qos_ea está programando la política en HW en la LC. Con **show feature-mgr client qos-ea policy summary location <node>** obtiene una descripción general de las políticas configuradas en una LC.

```
RP/0/RP0/CPU0:xrg-ncs-04#show feature-mgr client qos-ea policy summary location 0/0/CPU0
```

Mon Dec 4 22:03:12.371 UTC

Pol-Num Flags Classes Ref-cnt In-Hash Compiled Policy Name

19 0x00004000 2 4 Yes Yes **ingress_marking**

Si observa la vista detallada de este comando, puede ver dónde coincide en las clases:

```
RP/0/RP0/CPU0:xrg-ncs-04#show feature-mgr client qos-ea policy policy ingress_marking location 0/0/CPU0
```

Mon Dec 4 22:07:53.134 UTC

Policy name: **ingress_marking**

Policy Number: 19

Flags: 0x4000 [DNX_FM_POL_TOS]

Class Number: 2

Reference Count: 4

In Hash: Yes

Is Compiled: Yes

Level: 0 Class Number: 0 Class Name: i_match

Match Count: 2 Flags: 0x0 Logical Op: **MATCH_ANY**

ACE: 0 Class:0 Match:0 proto IPV4 Flags:0x8011 [Grant]

Source IPv4 Address: 0.0.0.0 (255.255.255.255)

Destination IPv4 Address: 0.0.0.0 (255.255.255.255)

Precedence Value: 3

UNKNOWN: UNKNOWN: UNKNOWN: ACE: 0 Class:0 Match:1 proto MPLS Flags:0x208040 [Grant]

MPLS EXP Value: 3

UNKNOWN: UNKNOWN: UNKNOWN: Final List

ACE: 0 Class:0 Match:0 proto IPV4 Flags:0x8011 [Grant]

Source IPv4 Address: 0.0.0.0 (255.255.255.255)

Destination IPv4 Address: 0.0.0.0 (255.255.255.255)

Precedence Value: 3

UNKNOWN: UNKNOWN: UNKNOWN: ACE: 1 Class:0 Match:1 proto MPLS Flags:0x208040 [Grant]

MPLS EXP Value: 3

UNKNOWN: UNKNOWN: UNKNOWN:

Level: 0 Class Number: 1 Class Name: **class-default**

Match Count: 1 Flags: 0x0 Logical Op: **MATCH_ANY**

ACE: 0 Class:1 Match:0 proto ANY Flags:0x28000 [Grant][Default]

UNKNOWN: UNKNOWN: UNKNOWN: Final List

ACE: 2 Class:1 Match:0 proto ANY Flags:0x28000 [Grant][Default]

UNKNOWN: UNKNOWN: UNKNOWN:

La información de QOS_EA también se puede mostrar desde la perspectiva del administrador de funciones. En primer lugar, se muestra el resumen:

```
RP/0/RP0/CPU0:xrg-ncs-04#show feature-mgr client qos-ea feature summary location 0/0/CPU0
```

```
Mon Dec 4 22:30:09.362 UTC
```

```
IFH NPU DIR Lookup-type ACL-ID Refcnt Feature-Name
```

```
-----
```

```
0x138 5 IN IPV4_QOS 14 1 ingress_marking:0
```

```
0x138 5 IN IPV6_QOS 14 1 ingress_marking:0
```

```
0x138 5 IN MPLS_QOS 14 1 ingress_marking:0
```

```
0x138 5 IN L2_QOS 14 1 ingress_marking:0
```

En el paso siguiente puede mostrar los detalles:

```
RP/0/RP0/CPU0:xrg-ncs-04#show feature-mgr client qos-ea feature feature-name ingress_marking:0
```

```
direction ingress lookup mpls interface HundredGigE 0/0/0/35 location 0/0/CPU0
```

```
Mon Dec 4 22:31:20.588 UTC
```

```
Feature name: ingress_marking:0
```

```
Lookup Type: MPLS_QOS
```

```
Direction: IN
```

```
Reference Count: 1
```

```
NPU: 5
```

```
Interfaces Attached: HundredGigE0_0_0_35
```

```
Number of Entries: 2
```

```
Number of Verified Entries: 0
```

```
Number of ACEs: 2
```

```
ACL ID: 14
```

```
ACL ID Ref Cnt: 4
```

```
ACL ID Min Length(bits): 0
```

```
ACL ID Max Length(bits): 5
```

```
ACL ID Num Entries: 30
```

```
ACL ID Num Free Entries: 29
```

```
Compression Level: 0 (UNCOMPRESSED)
```

El seguimiento qos_ea muestra el orden de los eventos cuando se aplica la política de servicio en una interfaz. Mostraría problemas en caso de que falle.

RP/0/RP0/CPU0:xrg-ncs-04#show qos ea trace all reverse location 0/0/CPU0

Mon Dec 4 22:17:26.297 UTC

3077 wrapping entries (36992 possible, 18816 allocated, 0 filtered, 3077 total)

Dec 4 21:30:18.280 qos_ea/int_evts 0/0/CPU0 t14563 CAPS: (line# 2882) : Responded to IM (batch size: 1)

Dec 4 21:30:18.280 qos_ea/int_evts 0/0/CPU0 t14563 CAPS: post_caps_processing : Registering with ETHER EA for subid notification. Bulk count 1

Dec 4 21:30:18.280 qos_ea/int_evts 0/0/CPU0 t14563 CAPS: caps_add_notify_remote Input num of interfaces 1

Dec 4 21:30:18.280 qos_ea/int_evts 0/0/CPU0 t14563 CAPS: process_caps_add : End: count = 1 ok_count 1. New num_policies_in 1 num_policies_out 2

Dec 4 21:30:18.280 qos_ea/int_evts 0/0/CPU0 t14563 SCOL: scol_interface_add : ifh 0x138

ea_intf_type MainIf : interface added to SCOL tree

Dec 4 21:30:18.280 qos_ea/int_evts 0/0/CPU0 t14563 SCOL: scol_interface_add_internal : ifh 0x138: after avl_tree_insert, new tree size is 1

Dec 4 21:30:18.280 qos_ea/int_evts 0/0/CPU0 t14563 UTIL: (line# 1665) Child info aggr bw add: Current propagated_bw_sum in npu_ctx after adding propagated_bw in if_qos 0 is 0, ifh 0x138

Dec 4 21:30:18.280 qos_ea/int_evts 0/0/CPU0 t14563 UTIL: insert_if_qos_from_list_by_npu_ctx : Inserting to list: if_qos 0x3087c0bb78 prev 0x0 next 0x0 ifh 0x138 head 0x0 tail 0x0 count 0

Dec 4 21:30:18.217 qos_ea/int_evts 0/0/CPU0 t14563 ACTN: results_add for ifh 0x138 with pmap ingress marking

Dec 4 21:30:18.217 qos_ea/int_evts 0/0/CPU0 t14563 CAPS: fm_program : npu_id 5 ifh 0x138

policy_name ingress_marking :Compiling policy and sending VMR to FM

Dec 4 21:30:18.217 qos_ea/int_evts 0/0/CPU0 t14563 ACTN: (line# 1597) : class_name class-default offset 0 policer_id 32: configuring policer...

Dec 4 21:30:18.217 qos_ea/int_evts 0/0/CPU0 t14563 POLICER-INT: policer_get : cir_kbps 720000000 cir_burst_kbits 32768 cir_burst 4194304pir_kbps 720000000 cir_burst_kbits 32768 pir_burst 4194304

Dec 4 21:30:18.217 qos_ea/int_evts 0/0/CPU0 t14563 POLICER-INT: policer_get : ifh 0x138 npu 5 offset 0 policer_id 32 mode/policer_type 0/0 inlif 0

Dec 4 21:30:18.217 qos_ea/int_evts 0/0/CPU0 t14563 POLICER-INT: policer_get : for ifh 0x138 for npu 5 offset 0

Dec 4 21:30:18.217 qos_ea/int_evts 0/0/CPU0 t14563 ACTN: (line# 1597) : class_name i_match offset 1 policer_id 33: configuring policer...

Dec 4 21:30:18.217 qos_ea/int_evts 0/0/CPU0 t14563 POLICER-INT: policer_get : cir_kbps 720000000 cir_burst_kbits 32768 cir_burst 4194304pir_kbps 720000000 cir_burst_kbits 32768 pir_burst 4194304

Dec 4 21:30:18.217 qos_ea/int_evts 0/0/CPU0 t14563 POLICER-INT: policer_get : ifh 0x138 npu 5 offset 1 policer_id 33 mode/policer_type 0/0 inlif 0

Dec 4 21:30:18.217 qos_ea/int_evts 0/0/CPU0 t14563 POLICER-INT: policer_get : for ifh 0x138 for npu 5 offset 1

Dec 4 21:30:18.217 qos_ea/int_evts 0/0/CPU0 t14563 ACTN: mark_actn_process Unconditional mark set for cmaps i_match

Dec 4 21:30:18.100 qos_ea/int_evts 0/0/CPU0 t14563 ACTN: resource_cfg for ifh 0x138 ul_ifh 0x0 with pmap ingress_marking

Dec 4 21:30:18.100 qos_ea/int_evts 0/0/CPU0 t14563 CMAP: classmap_process : chd 0x7feedd339b98, if_qos 0x3087c0bb78, class_info 0x3087c0b368

Dec 4 21:30:18.100 qos_ea/int_evts 0/0/CPU0 t14563 PWALK: preprocess_action : ifh 0x138 ul_ifh 0x0 policy_name ingress_marking class_name class-default

Dec 4 21:30:18.100 qos_ea/int_evts 0/0/CPU0 t14563 PWALK: pwalk_process_class : depth 0, usr_data->class_list 0x3087c0bb90, class_name class-default

Dec 4 21:30:18.100 qos_ea/int_evts 0/0/CPU0 t14563 CMAP: classmap_process : chd 0x7feedd33b478, if_qos 0x3087c0bb78, class_info 0x3087c0b770

Dec 4 21:30:18.100 qos_ea/int_evts 0/0/CPU0 t14563 PWALK: preprocess_action : ifh 0x138 ul_ifh 0x0 policy_name ingress_marking class_name i_match

Dec 4 21:30:18.100 qos_ea/int_evts 0/0/CPU0 t14563 PWALK: pwalk_process_class : depth 0, usr_data->class_list 0x3087c0bb90, class_name i_match

Dec 4 21:30:18.100 qos_ea/int_evts 0/0/CPU0 t14563 PWALK: pwalk_start_level : depth 0, stack 0x7fff8a11be40 ctx 0x7fff8a11c670

Dec 4 21:30:18.100 qos_ea/int_evts 0/0/CPU0 t14563 PWALK: policy_walk : policy_oper 0 policy_name ingress_marking ifh 0x138 ul_ifh 0x0 walk_flags 0x0

Dec 4 21:30:18.100 qos_ea/int_evts 0/0/CPU0 t14563 UTIL: if_qos_insert_to_db : Inserting to hash

```

table: npu_id 5 ifh 0x138, ul_ifh 0x0, dir 0
Dec 4 21:30:18.100 qos_ea/int_evts 0/0/CPU0 t14563 RATE: dnx_qosea_policy_resolve : policy_name
ingress_marking ifh 0x138 ul_ifh 0x0 intf_kbps 100000000
Dec 4 21:30:18.100 qos_ea/int_evts 0/0/CPU0 t14563 CAPS: policy_add : npu_id 5 ifh 0x138 ul_ifh
0x0 dir ingress policy_name ingress_marking num_classes 0 max_hierarchy 0
Dec 4 21:30:18.100 qos_ea/int_evts 0/0/CPU0 t14563 VRFY: policy_verify : Verifying policy
ingress_marking pmaphd 0x7feedd2c1088
Dec 4 21:30:18.100 qos_ea/int_evts 0/0/CPU0 t14563 UTIL: if_qos_init : Initializing if_qos
0x3087c0bb78 with npu_id 5 ifh 0x138 ul_ifh 0x0 direction ingress ea_intf_type MainIf
Dec 4 21:30:18.100 qos_ea/int_evts 0/0/CPU0 t14563 VRFY: check_policy_on_parent_subintf : Did
not find any reference to parent interface (0x138) in par_child_info_array
Dec 4 21:30:18.100 qos_ea/int_evts 0/0/CPU0 t14563 VRFY: check_policy_on_parent_subintf :
parent_ifh: 0x138, dir: 0 (ingress)
Dec 4 21:30:18.100 qos_ea/int_evts 0/0/CPU0 t14563 UTIL: intf_bw_get : Bandwidth of ifh 0x138 is
intf_kbps: 100000000 kbps
Dec 4 21:30:18.100 qos_ea/int_evts 0/0/CPU0 t14563 CAPS: caps_add_int : ifh 0x138 ea_intf_type
MainIf dir ingress
Dec 4 21:30:18.100 qos_ea/int_evts 0/0/CPU0 t14563 CAPS: caps_add : ifh 0x138 ea_intf_type
MainIf dir ingress policy_name ingress_marking
Dec 4 21:30:18.100 qos_ea/int_evts 0/0/CPU0 t14563 CAPS: process_caps_add : 1 of 1 direction
ingress on interface 0x138 policy_name ingress_marking
Dec 4 21:30:18.093 qos_ea/int_evts 0/0/CPU0 t14563 CAPS: process_caps_add : batch count = 1 .
Current num_policies_in 0 num_policies_out 2
Dec 4 21:30:18.093 qos_ea/int_evts 0/0/CPU0 t14563 INT: im_msg_handler : Rx: caps add message
(batch size: 1)

```

Cuando cambie la política y agregue, por ejemplo, un regulador a la clase `i_match`, estará visible en el seguimiento `qos_ea`. NCS5500 admite la modificación en el lugar de un mapa de políticas.

```

RP/0/RP0/CPU0:xrg-ncs-04(config-pmap-c)#police rate percent 50
RP/0/RP0/CPU0:xrg-ncs-04(config-pmap-c-police)#commit

```

```

Dec 4 22:49:39.567 qos_ea/int_evts 0/0/CPU0 t14563 POLICER-INT: policer_get : for ifh 0x138 for
npu 5 offset 0
Dec 4 22:49:39.567 qos_ea/int_evts 0/0/CPU0 t14563 POLICER-INT: (line# 147) : cir_kbps 50000000
cir_burst 4194304pir_kbps 0 pir_burst 0
Dec 4 22:49:39.567 qos_ea/int_evts 0/0/CPU0 t14563 POLICER-INT: (line# 144) : ifh 0x138 npu 5
offset 1 policer_type = 0 is_parent=0policer_id = 33
Dec 4 22:49:39.567 qos_ea/int_evts 0/0/CPU0 t14563 ACTN: (line# 1822) : class_name i_match
offset 0 policer_id 33: configuring policer...
Dec 4 22:49:39.567 qos_ea/int_evts 0/0/CPU0 t14563 ACTN: (line# 1597) : class_name i_match
offset 1 policer_id 33: configuring policer...

```

El comando `show qos interface` también refleja la adición:

```

RP/0/RP0/CPU0:xrg-ncs-04#show qos interface hundredGigE 0/0/0/35 input
Mon Dec 4 23:25:19.684 UTC
NOTE:- Configured values are displayed within parentheses
Interface HundredGigE0/0/0/35 ifh 0x138 -- input policy
NPU Id: 5
Total number of classes: 2
Interface Bandwidth: 100000000 kbps
Policy Name: ingress_marking
Accounting Type: Layer1 (Include Layer 1 encapsulation and above)
-----
Level1 Class = i_match
New qos group = 3
New discard class = 2

```

```
Policer Bucket ID = 0x21
Policer Stats Handle = 0x0
Policer committed rate = 49680000 kbps (50 %)
Policer conform burst = 4194304 bytes (default)
```

```
Level1 Class = class-default
```

```
Default Policer Bucket ID = 0x20
Default Policer Stats Handle = 0x0
Policer not configured for this class
```

El seguimiento qos_ea muestra también el valor convertido cuando se ejecuta el comando anterior.

El regulador utiliza un tamaño de paso incremental de 22 kbps. El valor configurado se redondea al múltiplo más cercano de 22 kbps cuando se utiliza en la fórmula de la tarjeta de línea.

```
Dec 4 23:25:19.773 qos_ea/int_evts 0/0/CPU0 t14563 POLICER-INT: policer_get : cir_kbps 49680000
cir_burst_kbits 32768 cir_burst 4194304pir_kbps 49680000 cir_burst_kbits 32768 pir_burst 4194304
Dec 4 23:25:19.773 qos_ea/int_evts 0/0/CPU0 t14563 POLICER-INT: policer_get : ifh 0x138 npu 5
offset 1 policer_id 33 mode/policer_type 0/0 inlif 0
Dec 4 23:25:19.773 qos_ea/int_evts 0/0/CPU0 t14563 POLICER-INT: policer_get : for ifh 0x138 for
npu 5 offset 1
```

QoS de salida

La QoS de salida admite acciones de colocación en cola. En NCS5500 se pueden tener dos políticas de servicio en la dirección de salida. Uno para marcar y otro para hacer cola. Esto es para una acción de marcado:

```
class-map match-any e_mark
match qos-group 3
end-class-map
!
```

```
policy-map egress_marking
class e_mark
set mpls experimental imposition 2
!
class class-default
!
end-policy-map
!
```

```
RP/0/RP0/CPU0:xrg-ncs-06#show feature-mgr client qos-ea policy summary location 0/0/CPU0
Thu Dec 21 10:44:33.978 UTC
Pol-Num Flags Classes Ref-cnt In-Hash Compiled Policy Name
-----
1 0x01000000 2 1 Yes Yes egress_marking
```

```
RP/0/RP0/CPU0:xrg-ncs-06#show feature-mgr client qos-ea feature summary location 0/0/CPU0
Thu Dec 21 10:59:08.872 UTC
IFH NPU DIR Lookup-type ACL-ID Refcnt Feature-Name
-----
0x340 0 OUT MAP_ID_QOS 0 1 egress_marking:0
```

Política de colas de egreso

```
class-map match-any tc1
  match traffic-class 1
end-class-map
!
class-map match-any tc2
  match traffic-class 2
end-class-map
!
class-map match-any tc3
  match traffic-class 3
end-class-map
!
class-map match-any tc4
  match traffic-class 4
end-class-map
!
class-map match-any tc5
  match traffic-class 5
end-class-map
!
policy-map POLICY_QOS_OUT
  class tc1
    priority level 1
    shape average percent 19
    queue-limit 500 us
  !
  class tc2
    priority level 2
    shape average percent 20
    queue-limit 500 us
  !
  class tc3
    bandwidth percent 1
  !
  class tc4
    bandwidth percent 20
  !
  class tc5
    bandwidth percent 20
  !
  class class-default
    bandwidth percent 20
  !
end-policy-map
!
```

One can quick verify where a QoS policy is applied.

```
RP/0/RP0/CPU0:xrg-ncs-04#show policy-map targets pmap-name POLICY_QOS_OUT location 0/0/CPU0
Mon Dec 18 21:13:01.463 UTC
1) Policymap: POLICY_QOS_OUT Type: qos
Targets (applied as main policy):
  HundredGigE0/0/0/35 output
Total targets: 1

Targets (applied as child policy):
Total targets: 0
```


El comando show QoS interface muestra los parámetros configurados y también los ID de cola para cada clase con una acción de colocación en cola configurada. Además, puede obtener valores predeterminados como el tamaño de la cola.

```
RP/0/RP0/CPU0:xrg-ncs-04#show qos interface hundredGigE 0/0/0/35 output
```

```
Thu Dec 14 20:56:29.796 UTC
```

```
NOTE:- Configured values are displayed within parentheses
```

```
Interface HundredGigE0/0/0/35 ifh 0x138 -- output policy
```

```
NPU Id: 5
```

```
Total number of classes: 6
```

```
Interface Bandwidth: 100000000 kbps
```

```
VOQ Base: 1032
```

```
Accounting Type: Layer1 (Include Layer 1 encapsulation and above)
```

```
-----  
Level1 Class (HP1) = tc1  
Egressq Queue ID = 1033 (HP1 queue)  
Queue Max. BW. = 0 kbps (19 %)  
Guaranteed service rate = 19000000 kbps  
TailDrop Threshold = 1187840 bytes / 500 us (500 us)  
WRED not configured for this class  
  
Level1 Class (HP2) = tc2  
Egressq Queue ID = 1034 (HP2 queue)  
Queue Max. BW. = 0 kbps (20 %)  
Guaranteed service rate = 20000000 kbps  
TailDrop Threshold = 1253376 bytes / 501 us (500 us)  
WRED not configured for this class  
  
Level1 Class = tc3  
Egressq Queue ID = 1035 (LP queue)  
Queue Max. BW. = 100884395 kbps (default)  
Queue Min. BW. = 1020015 kbps (1 %)  
Inverse Weight / Weight = 1 / (BWR not configured)  
Guaranteed service rate = 1000000 kbps
```

TailDrop Threshold = 1253376 bytes / 10 ms (default)

WRED not configured for this class

Level1 Class = tc4

Egressq Queue ID = 1036 (LP queue)

Queue Max. BW. = 100824615 kbps (default)

Queue Min. BW. = 20164923 kbps (20 %)

Inverse Weight / Weight = 1 / (BWR not configured)

Guaranteed service rate = 20000000 kbps

TailDrop Threshold = 25034752 bytes / 10 ms (default)

WRED not configured for this class

Level1 Class = tc5

Egressq Queue ID = 1037 (LP queue)

Queue Max. BW. = 100824615 kbps (default)

Queue Min. BW. = 20164923 kbps (20 %)

Inverse Weight / Weight = 1 / (BWR not configured)

Guaranteed service rate = 20000000 kbps

TailDrop Threshold = 25034752 bytes / 10 ms (default)

WRED not configured for this class

Level1 Class = class-default

Egressq Queue ID = 1032 (Default LP queue)

Queue Max. BW. = 100824615 kbps (default)

Queue Min. BW. = 20164923 kbps (20 %)

Inverse Weight / Weight = 1 / (BWR not configured)

Guaranteed service rate = 20000000 kbps

TailDrop Threshold = 25034752 bytes / 10 ms (default)

WRED not configured for this class

RP/0/RP0/CPU0:xrg-ncs-04#show feature-mgr client qos-ea policy summary location 0/0/CPU0

Thu Dec 14 21:27:30.486 UTC

Pol-Num Flags Classes Ref-cnt In-Hash Compiled Policy Name

3	0x00010000	6	0	Yes	No POLICY_QOS_OUT
2	0x00000000	1	0	Yes	No p_out
1	0x00004000	2	4	Yes	Yes ingress_marking

RP/0/RP0/CPU0:xrg-ncs-04#show feature-mgr client qos-ea policy policy POLICY_QOS_OUT location 0/0/CPU0

Thu Dec 14 21:28:22.182 UTC

Policy name: POLICY_QOS_OUT

Policy Number: 3

Flags: 0x10000 [DNX_FM_POL_TC]

Class Number: 6

Reference Count: 0

In Hash: Yes

Is Compiled: No

Level: 0 Class Number: 0 Class Name: tc1

Match Count: 1 Flags: 0x0 Logical Op: MATCH_ANY

ACE: 0 Class:0 Match:0 proto ANY Flags:0x8080 [Grant]

DNX_FM_RANGE_TC: (1, 1)

UNKNOWN: UNKNOWN:

Level: 0 Class Number: 1 Class Name: tc2

Match Count: 1 Flags: 0x0 Logical Op: MATCH_ANY

ACE: 0 Class:1 Match:0 proto ANY Flags:0x8080 [Grant]

DNX_FM_RANGE_TC: (2, 2)

UNKNOWN: UNKNOWN:

Level: 0 Class Number: 2 Class Name: tc3

Match Count: 1 Flags: 0x0 Logical Op: MATCH_ANY

ACE: 0 Class:2 Match:0 proto ANY Flags:0x8080 [Grant]

DNX_FM_RANGE_TC: (3, 3)

UNKNOWN: UNKNOWN:

Level: 0 Class Number: 3 Class Name: tc4

Match Count: 1 Flags: 0x0 Logical Op: MATCH_ANY

ACE: 0 Class:3 Match:0 proto ANY Flags:0x8080 [Grant]

DNX_FM_RANGE_TC: (4, 4)

UNKNOWN: UNKNOWN:

Level: 0 Class Number: 4 Class Name: tc5

Match Count: 1 Flags: 0x0 Logical Op: MATCH_ANY

ACE: 0 Class:4 Match:0 proto ANY Flags:0x8080 [Grant]

DNX_FM_RANGE_TC: (5, 5)

UNKNOWN: UNKNOWN:

Level: 0 Class Number: 5 Class Name: class-default

Match Count: 1 Flags: 0x0 Logical Op: MATCH_ANY

ACE: 0 Class:5 Match:0 proto ANY Flags:0x28000 [Grant][Default]

UNKNOWN: UNKNOWN:

Adding WRED:

```
policy-map POLICY_QOS_OUT
class tc4
random-detect discard-class 1 5 ms 15 ms
!
class class-default
!
end-policy-map
!
end
```

```
Level1 Class = tc4
Egressq Queue ID = 1036 (LP queue)
Queue Max. BW. = 100824615 kbps (default)
Queue Min. BW. = 20164923 kbps (20 %)
Inverse Weight / Weight = 1 / (BWR not configured)
Guaranteed service rate = 20000000 kbps
TailDrop Threshold = 75497472 bytes / 30 ms (default)
```

WRED profile for **Discard_Class 1**

```
WRED Min. Threshold = 12517376 bytes (5 ms)
WRED Max. Threshold = 37748736 bytes (15 ms)
```

Default RED profile

```
WRED Min. Threshold = 0 bytes (0 ms)
WRED Max. Threshold = 0 bytes (0 ms)
```

5 location 0/0/CPU0

Tue Nov 7 17:24:03.570 UTC

Node ID: 0/0/CPU0

Intf name	Intf handle (hex)	NPU #	NPU core	PP Port	Sys Port	VOQ base	Flow base	VOQ port type	Port speed (Gbps)
-----------	-------------------	-------	----------	---------	----------	----------	-----------	---------------	-------------------

Hu0/0/0/35	138	5	0	1	321	1032	1064	local	100	<- 1032 (Default LP queue) - displays the base queue -> refer to show qos interface commaand.
------------	-----	---	---	---	-----	------	------	-------	-----	---

Display the queue details:

RP/0/RP0/CPU0:xrg-ncs-04#show controllers fia diagshell 5 "diag cosq voq id=1036 det=1" location 0/0/CPU0 <- 1036 (LP queue) - refer to show qos interface commaand.

Thu Dec 14 21:51:33.184 UTC

Node ID: 0/0/CPU0

Core 0:

Basic info

Q type: voq

num cos: 8, cosq class: 4

Base queue id: 1032, base queue gport: 0x24000408

Credit request type: BCM_COSQ_DELAY_TOLERANCE_100G_SLOW_ENABLED Adjusted for slow enabled 100Gb ports

Watchdog enable in common status message mode

Is queue in credit watchdog queue range:True

Credit watchdog message time: 0

Delete queue time: 512

Backoff enter queue credit balance threshold: 40960

Backoff exit queue credit balance threshold: 40960

Backlog enter queue credit balance threshold: 40960

Backlog exit queue credit balance threshold: 40960

Empty queue satisfied credit balance threshold: 0

Max empty queue credit balance threshold: 16384

Exceed max empty queue credit balance threshold: 1

Off-To-Slow credit balance threshold: 0

Off-To-Normal credit balance threshold: 38912

Slow-To-Normal credit balance threshold: 38912

Normal-To-Slow credit balance threshold: 38912

Delay Tolerance is NOT OCB only

Delay Tolerance is NOT High Q Priority

Slow Level Thresh Down in slow level number: 0 is: 0

Slow Level Thresh Down in slow level number: 1 is: 0

Slow Level Thresh Down in slow level number: 2 is: 0

Slow Level Thresh Down in slow level number: 3 is: 0

Slow Level Thresh Down in slow level number: 4 is: 0

Slow Level Thresh Down in slow level number: 5 is: 0

Slow Level Thresh Down in slow level number: 6 is: 0

Slow Level Thresh Up in slow level number: 0 is: 0

Slow Level Thresh Up in slow level number: 1 is: 0

Slow Level Thresh Up in slow level number: 2 is: 0

Slow Level Thresh Up in slow level number: 3 is: 0

Slow Level Thresh Up in slow level number: 4 is: 0

Slow Level Thresh Up in slow level number: 5 is: 0

Slow Level Thresh Up in slow level number: 6 is: 0

Credit value(local): 1024, Credit value(remote): 2048

Credit discount value: -2

Rate class info

WRED info:

Green : enable(False) min_thresh(0) max_thresh(0) drop_probability(0)

**Yellow: enable(True) min_thresh(12517376) max_thresh(37748736)
drop_probability(9)**

Red : enable(False) min_thresh(0) max_thresh(0) drop_probability(0)

Black : enable(False) min_thresh(0) max_thresh(0) drop_probability(0)

Tail drop info:

Green : max queue size in bytes(75497472), max queue size in BDs(294912)

Yellow: max queue size in bytes(75497472), max queue size in BDs(294912)

Red : max queue size in bytes(75497472), max queue size in BDs(294912)

Black : max queue size in bytes(75497472), max queue size in BDs(294912)

Guaranteed info:

Green : min queue size in bytes(0), min queue size in BDs(0)

Yellow: min queue size in bytes(0), min queue size in BDs(0)

Red : min queue size in bytes(0), min queue size in BDs(0)

Black : min queue size in bytes(0), min queue size in BDs(0)

Fair adaptive tail drop info:

Enable: True

Green : alpha(-2)

Yellow: alpha(-2)

Red : alpha(-2)

Black : alpha(-2)

ECN wred info: enable(False) min_thresh(0) max_thresh(0) drop_probability(0)

ECN max queue size in bytes(524288), max queue size in BDs(524288)

VSQ-related

Category class: 2

Traffic class: 4

Connection class: 0

Green : Admission-test-templates(0)

Yellow: Admission-test-templates(0)

Red : Admission-test-templates(0)

Black : Admission-test-templates(0)

AdmissionTestProfileA[0]: 'category, traffic class'

AdmissionTestProfileB[0]: 'category'

current queue size: 0 bytes, current queue bds size: 0

Attached VOQ connector: 0x00000428

Destination sys port: 0x00007fff

OCB eligiblity: True

[SNIP]

output for core 1 is ommitted.

Para monitorear el tamaño de cola puede ejecutar el comando anterior y grep para la línea correspondiente.

Ejemplo: **show controllers fia diagshell 5 "diag cosq voq id=1036 core=0 det=1" location 0/0/CPU0 | i tamaño de cola actual**

Las estadísticas se pueden mostrar con el comando **show policy-map interface**.

Ejemplo: **show policy-map interface cientos GigE 0/0/0/35 output location 0/0/CPU0**

Nota: El resultado no se detalla como lo conocemos de otras plataformas XR. No muestra líneas para el perfil WRED o RED correspondiente.

```
RP/0/RP0/CPU0:xrg-ncs-04#show policy-map interface hundredGigE 0/0/0/35 output location 0/0/CPU0
```

```
Thu Dec 14 22:05:50.971 UTC
```

```
Interface:HundredGigE0/0/0/35 Location: node0_0_CPU0 output: POLICY_QOS_OUT
```

```
Class tc1
```

Classification statistics	(packets/bytes)	(rate - kbps)
Matched :	0/0	0
Transmitted :	0/0	0
Total Dropped :	0/0	0

```
Queueing statistics
```

Queue ID	: 1033
Taildropped(packets/bytes)	: 0/0

```
Class tc2
```

Classification statistics	(packets/bytes)	(rate - kbps)
Matched :	0/0	0
Transmitted :	0/0	0

Total Dropped : 0/0 0

Queueing statistics

Queue ID : 1034

Taildropped(packets/bytes) : 0/0

Class tc3

Classification statistics (packets/bytes) (rate - kbps)

Matched : 0/0 0

Transmitted : 0/0 0

Total Dropped : 0/0 0

Queueing statistics

Queue ID : 1035

Taildropped(packets/bytes) : 0/0

Class tc4

Classification statistics (packets/bytes) (rate - kbps)

Matched : 0/0 0

Transmitted : 0/0 0

Total Dropped : 0/0 0

Queueing statistics

Queue ID : 1036

Taildropped(packets/bytes) : 0/0

Class tc5

Classification statistics (packets/bytes) (rate - kbps)

Matched : 0/0 0

Transmitted : 0/0 0

Total Dropped : 0/0 0

Queueing statistics

Queue ID : 1037

Taildropped(packets/bytes) : 0/0

Class class-default

Classification statistics (packets/bytes) (rate - kbps)

Matched : 0/0 0

Transmitted : 0/0 0

Total Dropped : 0/0

Queueing statistics

Queue ID : 1032

Taildropped(packets/bytes) : 0/0

Supervisar los FMQ

El tráfico de multidifusión no está programado. Utiliza colas de multidifusión de fabric (FMQ) en lugar de VOQ convencionales. En esa puntuación, salida multicast, QoS no se soporta.

Para mostrar las estadísticas de los FMQs, este comando está disponible a partir de 6.5.2:

```
RP/0/RP0/CPU0:NCS5508#show controllers npu stats voq base 0 instance 0 location 0/0/CPU0  
Mon Jul 8 08:59:22.465 UTC
```

Asic Instance = 0

VOQ Base = 0

	ReceivedPkts	ReceivedBytes	DroppedPkts	DroppedBytes
TC_0 = 224		15673	49	3468
TC_1 = 0		0	0	0
TC_2 = 0		0	0	0
TC_3 = 0		0	0	0
TC_4 = 0		0	0	0
TC_5 = 0		0	0	0
TC_6 = 0		0	0	0
TC_7 = 0		0	0	0

También es posible seguir un paquete a través de la NPU y comprobar si, por ejemplo, se aplica la clase de tráfico (TC) adecuada. Eche un vistazo a

Sección **Siga un paquete de tránsito** para los comandos y el filtro del shell de diagnóstico correspondientes.

Comandos adicionales a recopilar

Estos son algunos comandos adicionales:

- **show tech-support qos pi**
- **show tech-support qos platform**
- **show tech cef platform**

Si se sospecha un problema en una interfaz individual:

- **show run interface <name>**
- **show run policy-map <>** (para los policy-maps aplicados en la interfaz)
- **show qos interface <name> input|output**
- **show policy-map interface <name>**
- **show qos remote location <node>** (para política de salida)

Si la interfaz es un paquete, proporcione información completa de miembro del paquete.

- **show bundle <>**

Limitaciones/advertencias

Marcación

- Las marcas L2(COS, DEI) y L3(MPLS EXP) no se soportan en el mismo policy-map
- El regulador no funciona cuando se combina con **set discard-class**.
- match qos-group es sólo para marcar.
- qos-group 0 está reservado para la clase predeterminada.
- las estadísticas y los contadores no se muestran para una política de marcado.

Ancho de banda

- La suma de todas las sentencias de **ancho de banda** en un policy-map no debe ser > 100%
- La suma de todas las sentencias **restantes de ancho de banda** en un policy-map no debe ser > 100%
- Un modelador no se comparte entre las clases y, por lo tanto, no hay restricciones en el resumen de las sentencias de ancho de banda.

QoS de ingreso

Las acciones de cola no se admiten en una política de servicio de ingreso.

QoS de salida

- Las capacidades/acciones de marcación son muy limitadas.
- Solo se admite una coincidencia de política de salida en clase de tráfico (TG), qos-group (QG) y clase de descarte (DC). TC - para acciones de colocación en cola QG: para marcar acciones DC - WRED
- Una política de servicio que conecte el siguiente policy-map sería rechazada en una interfaz de salida:

```
policy-map SET-AND-QUEUE-OUT
  class GOLD
    priority level 1
    shape average percent 20
    queue-limit 500 us
    set mpls experimental imposition 5
```

El mensaje de error que recibe es: !!% "DNX_QOSEA" ha detectado la condición de 'advertencia' "Configuración de mapa de políticas de salida no válida. Utilice qos-group para marcar las funciones y la clase de tráfico para las funciones de envío a cola".

Es posible aplicar dos políticas de servicio independientes, una para la marcación y otra para la colocación en cola.

HQoS (calidad de servicio jerárquica)

- Para aplicar una política de servicio en una subinterfaz, se requiere el perfil HQoS

independiente si es plano.

- Aplicar una política de servicio en una subinterfaz que contiene una clase de prioridad requiere un modelador en esta clase independiente si es jerárquico o plano.
- En el modo HQoS, las clases de prioridad deben tener una configuración de límite de velocidad (modelador). El valor efectivo del modelador se toma como una reserva de ancho de banda prioritaria.
- En el modo HQoS, el ancho de banda (BW) y el ancho de banda restante (BWR) no se admiten simultáneamente en la misma configuración de mapa de políticas. El ancho de banda o el ancho de banda restante se pueden configurar en el mismo policy-map.
- BW respectivo BWR junto con la forma no se soporta en el mismo policy-map antes de la versión 6.3.1.
- WRED no está permitido en la clase primaria
- Se admiten un máximo de 4 niveles de prioridad en el modo HQoS
- La regulación de HQoS de entrada y el perfil de peering de entrada no se soportan juntos.
- Para la política de dos niveles, no se admiten las clases definidas por el usuario en el nivel primario.
- Debido a una limitación TCAM, la selección de túnel basada en políticas (PBTS) no funciona cuando el perfil HQoS está habilitado.

Advertencia de configuración no admitida después de la confirmación

En algunos casos, puede ver un registro de advertencia después de confirmar una configuración de QoS como en el siguiente ejemplo:

% Advertencias de validación detectadas como resultado de la operación de confirmación.
Ejecute 'show configuration warnings' para ver las advertencias

Si ejecuta la CLI sugerida, verá:

```
!! SEMANTIC ERRORS: This configuration was rejected by
!! the system due to semantic errors. The individual
!! errors with each failed configuration command can be
!! found below.
```

<Aquí se imprime el motivo>

Sin embargo, se aplica la configuración. En este caso, podría proceder de la verificación de Yang.

Compruebe si está configurada la **validación de la configuración**.

Si este es el caso, es probable que se haya despedido una regla de yang.

Puede activar **debug configuration Validation ccv detail** y volver a confirmar la configuración.

Debería ver la regla yang, que está activando el mensaje "No admitido" en el resultado de la depuración.

Example:

RP/0/RP0/CPU0:Jan 22 15:57:44.441 UTC: ccv[189]: ...essage : 'Service Policy on interface "TenGigE0/0/0/7/1.102" is not supported', rule group name: 'qos-ma', rule name: '**cac-ser-pol-qos-in-unsupp-intf**').

Póngase en contacto con el equipo de IP de QoS en este caso.