

# Informazioni sull'allocazione dei buffer di coda sugli switch Catalyst 9000

## Sommario

---

[Introduzione](#)

[Prerequisiti](#)

[Requisiti](#)

[Componenti usati](#)

[Prodotti correlati](#)

[Premesse](#)

[Terminologia](#)

[Buffer](#)

[Cambia algoritmo allocazione buffer coda](#)

[Configurazione](#)

[Scenario 1: Determinazione del valore di base del buffer per una determinata velocità della porta](#)

[Scenario 2: Determinazione del buffer soft e hard per più code e rapporti espliciti](#)

[Scenario 2.1: Modifica moltiplicatore configurato dall'utente dallo scenario 2](#)

[Scenario 3: Determinare il buffer soft e hard per più code e un rapporto implicito](#)

[Scenario 3.1: Determinazione del buffer soft e hard per più code e rapporti impliciti multipli](#)

[Scenario 4: La somma del rapporto buffer coda tra le code è inferiore a 100](#)

[Scenario 5: Determinazione del soft buffer e del buffer di una classe con priorità di livello 2](#)

[Limite coda](#)

[Informazioni correlate](#)

---

## Introduzione

Questo documento descrive come prevedere l'allocazione del buffer delle code alle code del traffico sugli switch Catalyst serie 9000.

## Prerequisiti

### Requisiti

Cisco raccomanda la conoscenza dei seguenti argomenti:

- Concetti e processi QoS (Quality of Service) per le modalità di contrassegno, coda e pianificazione dei pacchetti
- Configurazione QoS Cisco MQC

## Componenti usati

Le informazioni fornite in questo documento si basano sulle seguenti versioni software e hardware:

- Cisco Catalyst 9500-25Y4C
- Cisco IOS® XE 17.4.1

Le informazioni discusse in questo documento fanno riferimento a dispositivi usati in uno specifico ambiente di emulazione. Su tutti i dispositivi menzionati nel documento la configurazione è stata ripristinata ai valori predefiniti. Se la rete è operativa, valutare attentamente eventuali conseguenze derivanti dall'uso dei comandi.

## Prodotti correlati

Il presente documento può essere utilizzato anche per le seguenti versioni hardware e software:

- Catalyst serie 9200 - 9600 Switch
- Catalyst 9300X e 9400X
- Tutte le versioni del software Cisco IOS XE



Nota: Il presente documento non si applica agli switch 9500X o 9600X poiché utilizzano un'architettura ASIC e QoS diversa.

---

## Premesse

Per una panoramica tecnica della funzionalità QoS sugli switch Catalyst serie 9000, vedere: [White paper su Catalyst 9000 QoS e code.](#)

Spesso è necessario regolare l'allocazione dei buffer in risposta a cali di output indesiderati per una particolare classe di traffico. Per ulteriori informazioni su come diagnosticare e risolvere i problemi di perdite di output sugli switch Catalyst serie 9000, fare riferimento a questo articolo: [Risoluzione dei problemi relativi ai pacchetti eliminati nella coda di output sui dispositivi Catalyst](#)

## Terminologia

QoS	Quality of Service (QoS)	Concetto/gruppo di funzionalità correlate relative alla classificazione, alla contrassegno, alla coda e alla pianificazione del traffico in entrata e in uscita da un dispositivo di rete
DSCP	Valore DSCP	Meccanismo di classificazione del traffico contenuto nell'intestazione IP di un pacchetto
CoS	Class of Service	Meccanismo di classificazione del traffico contenuto nell'intestazione frame Ethernet di un pacchetto
ASSO	Voce di controllo di accesso	Una singola regola o riga in un Access Control List (ACL)
ACL	Lista dei controlli di accesso	Gruppo di voci di controllo di accesso (ACE, Access Control Entries) utilizzate da diverse funzionalità per associare il traffico e intraprendere un'azione
ASIC	Circuito integrato specifico dell'applicazione	Chip per computer progettato per eseguire un'attività o una serie di attività specifiche con efficienza elevata.
UADP	Unified Access Data Plane	Cisco ASIC è stato usato negli switch Catalyst serie 9000 per eseguire molte attività di elaborazione dei pacchetti di rete.
PBC	Complesso buffer di pacchetto	Sottosistema Cisco UADP ASIC, che funge da buffer di pacchetto centrale per elaborare, mettere in coda e pianificare i pacchetti.
AQM	Gestione code attive	Sottosistema Cisco UADP ASIC che gestisce le code del traffico e pianifica le azioni per le porte di rete.
DTS	Soglia e scala dinamiche	Tecnologia Cisco UADP ASIC che adatta e adatta dinamicamente i buffer su più porte per ottimizzare l'utilizzo dell'hardware

## Buffer

In teoria, i buffer sono memoria utilizzata per assorbire burst transitori di dati, quando i dati scambiati o instradati a una porta superano la capacità delle porte di caricare i dati sul cavo. Una porta ha una velocità fissa alla quale trasmette e rimuove i dati dalla coda. Un buffer, a livello concettuale, è semplicemente un luogo in cui memorizzare, o accodare, i dati fino a quando non vengono trasmessi fuori dall'interfaccia.

Sugli switch Catalyst serie 9000, la parola buffer può essere utilizzata in due modi. Il buffer di sistema nel suo complesso è noto anche come PBC (Packet Buffer Complex) dell'ASIC. La parola buffer può anche riferirsi a una piccola unità della PBC. Un buffer viene allocato alle porte in base alla coda. In altre parole, a una coda di porta viene allocata una quantità di singoli buffer di piccole dimensioni dal buffer di sistema complessivo.

Sulle piattaforme Cisco UADP basate su ASIC, un buffer contiene fino a 256 byte di dati e i buffer sono collegati tra loro per rappresentare frame più grandi di 256 byte.

## Cambia algoritmo allocazione buffer coda

- A una porta è assegnato un valore di buffer di base hardcoded che può essere scalato dall'utente o dal sistema (ma il numero di base non può essere modificato, il che è importante per capire come prevedere i risultati)
- Il valore di base del buffer per un determinato tipo di porta dipende dalla combinazione specifica di hardware (modello di switch), versione del software e velocità della porta. Ad esempio, una porta Gigabit Ethernet su Catalyst C9500-25Y4C inizia con un valore del buffer di base diverso da quello di una porta Gigabit Ethernet su Catalyst 9200L
- Questo valore di base viene suddiviso in N numero di code, in base alla configurazione.

Il calcolo finale del buffer disponibile per coda è influenzato da questi fattori:

- Numero di code configurate
- Il rapporto code-buffer, un comando che consente di regolare la modalità di suddivisione del buffer di base tra le code di porte configurate.
- Moltiplicatore di buffer di base configurato dall'utente (configurato come qos queue-softmax-multiplier <percentuale>)
- Moltiplicatore di base del buffer definito dal sistema (non configurabile dall'utente, 4x o 400%). Sebbene questa impostazione sia applicata per impostazione predefinita, in alcune circostanze la relativa rimozione rende importante comprendere questo valore 4x per prevedere l'algoritmo di allocazione

- Entrambi i moltiplicatori influiscono solo sui cosiddetti buffer soft

I soft buffer sono buffer condivisi tra più porte. Questi buffer sono chiamati soft perché non sono garantiti per la porta.

Il sistema sovrascrive intenzionalmente i soft buffer. Ciò consente a una porta di utilizzare un numero elevato di buffer, se necessario, ma poiché più porte richiedono buffer, tutte le porte e le code vengono ridotte in modo dinamico e corretto come parte del processo Cisco UADP ASIC DTS.

In sintesi, i soft buffer - definiti negli output come softmax, sono un valore massimo opportunistico. Una porta utilizza il valore softmax completo solo se la quantità di buffer è disponibile nel buffer di sistema complessivo. Quando la richiesta di buffer aumenta attraverso altre porte e code, il buffer massimo disponibile per la porta è inferiore.

I buffer hardware sono buffer riservati in modo esplicito a una porta e non sono influenzati dal processo DTS. Poiché i buffer rigidi sono garantiti, la somma dei buffer rigidi totali allocati alle porte non supera mai il segmento PBC dedicato a questi buffer rigidi.

I meccanismi che governano la scala attiva del soft buffer sono noti come DTS (Dynamic Threshold and Scale), descritti nel [white paper](#) sulla [QoS di Catalyst serie 9000](#).

Le dimensioni dei segmenti PBC dedicati ai buffer hardware e soft della coda cambiano in modo dinamico durante la configurazione del sistema e possono essere visualizzati come AQM GlobalSoftLimit e GlobalHardLimit in questo output:

```
<#root>
```

```
C9500#
```

```
show platform hardware fed active qos queue stats interface twel/0/1
```

```
-----  
AQM Global counters  
GlobalHardLimit:
```

```
18072
```

```
| GlobalHardBufCount: 0  
GlobalSoftLimit:
```

```
37224
```

```
| GlobalSoftBufCount: 0
```

C9500#

show platform hardware fed active qos queue config interface twl/0/1

Asic:0 Core:1 DATA Port:20 GPN:101 LinkSpeed:0x12  
AFD:Disabled FlatAFD:Disabled QoSMap:0 HW Queues: 160 - 167  
DrainFast:Disabled PortSoftStart:2 - 4320 BufferSharing:Disabled  
DTS

Hardmax

Softmax

PortSMin	GlbISMin	PortStEnd	QEnable
0	1	2	
480			
3			
1920			
16	960	0	0
4	5760		En

<--- default configuration has a mix of hard buffer and soft buffer in queue 0

1	1	0
0		

4		
2880		

16	1440	8	720	4	5760	En
----	------	---	-----	---	------	----

<--- default configuration has two queues so some buffers are seen in queue 1

C9500(config)#

```
policy-map test
```

```
C9500(config-pmap)#
```

```
class class-default
```

```
C9500(config-pmap-c)#
```

```
priority level 1
```

```
<--- Priority level 1 queue configuration on first queue, which is queue 0 in the next output
```

```
C9500(config-pmap-c)#
```

```
exit
```

```
C9500(config-pmap)#
```

```
exit
```

```
C9500(config)#
```

```
int tw1/0/1
```

```
C9500(config-if)#
```

```
service-policy output test
```

```
C9500(config-if)#
```

```
end
```

```
C9500#
```

```
show platform hardware fed active qos queue config interface twel/0/1
```

```
Asic:0 Core:1 DATA Port:20 GPN:101 LinkSpeed:0x12  
AFD:Disabled FlatAFD:Disabled QoSMap:0 HW Queues: 160 - 167  
  DrainFast:Disabled PortSoftStart:4 - 1800 BufferSharing:Disabled  
  DTS
```

Hardmax

Softmax

```
PortSMin GblSMin PortStEnd QEnable
-----
0   1  4
1200
7
1200
0   0 0   0 3  2400  En
```

<--- Hardmax increased to 1200 from 480 in queue 0, softmax reduced to 1200 from 1920

```
1   1 0   0 0   0 0   0 0   0 3  2400  En
```

<--- queue 1 now no longer has any values, as no second queue is configured

C9500#

show platform hardware fed active qos queue stats interface twel/0/1

```
-----
AQM Global counters
GlobalHardLimit:
```

18792

| GlobalHardBufCount: 0

<--- GlobalHardLimit increased to 18792 from 18072, or by 720

GlobalSoftLimit:

36504

| GlobalSoftBufCount: 0

<---

GlobalSoftLimit decreased from 37224 to 36504, or by 720



Nota: Si noti l'aumento di GlobalHardLimit e la riduzione proporzionale di GlobalSoftLimit.

---

Inoltre, quando si configura il livello di priorità 1, il softmax per quella coda viene impostato in modo statico in modo da essere esattamente uguale al hardmax. È possibile modificare solo il buffer fisso per una coda con livello di priorità 1.

La modifica in GlobalHardLimit e GlobalSoftLimit è uguale a 720. Ciò equivale anche alla modifica in hardmax dopo la configurazione.

Gli scenari illustrati in questo documento spiegano come calcolare e prevedere le allocazioni softmax e hardmax in più configurazioni di mappe delle policy.

## Configurazione

### Scenario 1: Determinazione del valore di base del buffer per una determinata velocità della porta

Un valore di buffer finale delle code è in parte una funzione di un valore di base che viene prima allocato tra le code. Questo valore viene quindi moltiplicato in seguito nel caso di buffer soft.

I fattori di moltiplicazione, in combinazione con altri comportamenti impliciti, rendono la determinazione di un valore finale per una determinata coda, con una determinata configurazione, una sfida.

Il primo passaggio per chiarire l'allocazione del buffer della coda risultante consiste nel determinare il valore del buffer di base.

A tale scopo, utilizzare una coda di priorità, che riceve il buffer hardware direttamente proporzionale al numero di code o al rapporto di buffer della coda configurato.

Con una configurazione specifica, è possibile derivare in modo esplicito la quantità di buffer di

base allocata a una determinata velocità della porta.

Configurare e assegnare tutto il buffer a una singola coda non moltiplicata (una coda di livello di priorità 1)

In questo esempio, la classe class-default viene utilizzata per far corrispondere tutto il traffico, poiché non sono configurate altre classi.

```
<#root>
Switch(config)#

policy-map test1

Switch(config-pmap)#

class class-default

Switch(config-pmap-c)#

priority level 1

<--- Assign hard buffer to the port, which is not affected by multipliers

Switch(config-pmap-c)#

queue-buffers ratio 100

<--- Assign all buffers to this queue only
```

La configurazione dell'esempio precedente effettua le seguenti azioni:

- Il livello di priorità 1 indica al sistema di eseguire una pianificazione rigida per questa coda/classe e allocare il buffer rigido a questa coda/classe.
  - La pianificazione dei pacchetti esula dall'ambito di questo documento, ma in genere una coda rigida viene servita prima di tutte le altre.
- Il rapporto buffer coda 100 alloca 100/100 o il 100% del buffer di base disponibile per questa

coda/classe.

In una mappa dei criteri con più classi, non è possibile allocare il 100% del buffer a una singola classe. È necessario allocare almeno 1/100 o 1% a qualsiasi classe.

In un criterio con una sola classe, è possibile allocare tutto il buffer ad essa.

Come indicato in precedenza, una coda con priorità ottiene i buffer hardware con la distribuzione del buffer di base in base al rapporto di coda-buffer configurato. Un buffer fisso non è soggetto ad alcun moltiplicatore.

L'hard buffer è osservato negli output sotto una colonna intitolata Hardmax.

Ora si dispone di una singola classe di traffico con buffer non soggetti ad alcun moltiplicatore. In questo modo, è possibile derivare esplicitamente l'allocazione del buffer di base per questa velocità della porta (e solo questa velocità della porta su questa piattaforma, altre sono diverse), perché il buffer di base e il valore hardmax sono uguali.

Buffer di base = ?

Rapporto coda 1 =  $100/100 = 1$

Hardmax per questa coda = Buffer base x Rapporto coda 1

$X = Y \times 1$

$X / 1 = Y$

$X = Y$

$X = Y = \text{Hardmax} = \text{buffer di base} = 1200$  (vedere Esempio 2).

In questo esempio, il test1 della mappa dei criteri viene applicato a un'interfaccia come criterio del servizio di output

<#root>

9500H(config)#

int tw1/0/3

```
9500H(config-if)#
```

```
service-policy output test1 <--- service policy that assigns all buffer to the first
```

```
9500H#
```

```
show platform hardware fed active qos queue config interface tw1/0/3
```

```
Asic:0 Core:1 DATA Port:22 GPN:103 LinkSpeed:0x12  
AFD:Disabled FlatAFD:Disabled QoSMap:0 HW Queues: 176 - 183  
DrainFast:Disabled PortSoftStart:4 - 1800 BufferSharing:Disabled  
DTS
```

```
Hardmax
```

```
Softmax PortSMin GblSMin PortStEnd QEnable  
-----  
0 1 4
```

```
1200
```

```
7 1200 0 0 0 0 3 2400 En
```

```
<--- hardmax 1200 - the maximum amount of buffer this port can use without multiplication
```

```
1 1 0 0 0 0 0 0 0 0 3 2400 En  
2 1 0 0 0 0 0 0 0 0 3 2400 En  
3 1 0 0 0 0 0 0 0 0 3 2400 En  
4 1 0 0 0 0 0 0 0 0 3 2400 En  
5 1 0 0 0 0 0 0 0 0 3 2400 En  
6 1 0 0 0 0 0 0 0 0 3 2400 En  
7 1 0 0 0 0 0 0 0 0 3 2400 En
```

```
<snip>
```

Come illustrato, il valore hardmax per questa coda di priorità con il 100% del buffer allocato è 1200.

Poiché hardmax è un valore completamente non moltiplicato/non scalato e il 100% del buffer è configurato per questa coda, l'allocazione base del buffer per questo modello specifico di switch, versione software e velocità specifica della porta è 1200.

Le altre velocità delle porte su questo stesso switch e su altri modelli di switch alla stessa velocità delle porte ricevono allocazioni del buffer di base diverse. Questa allocazione di base non è configurabile dall'utente e deve essere derivata tramite osservazione.

Per gli scenari riportati di seguito, l'uso dello switch, del software e della velocità delle porte è lo

stesso. Quindi, tutti presuppongono un'allocazione di base di 1200 per i calcoli per determinare l'allocazione finale del buffer.



Nota: Anche il valore softmax dell'esempio precedente è 1200.

---

In base alla progettazione, una coda di livello di priorità 1 ha softmax uguale esattamente al suo hardmax. Questa opzione è destinata e non configurabile dall'utente.

Inoltre, questo caso specifico di allocazione softmax non è influenzato dai moltiplicatori softmax mostrati più avanti. Solo una coda di livello di priorità 1 presenta questo comportamento per softmax, che è previsto.

## Scenario 2: Determinazione del buffer soft e hard per più code e rapporti espliciti

In questo scenario, viene aggiunta una coda aggiuntiva. Questa coda non utilizza il livello di priorità 1, quindi softmax si adatta con i moltiplicatori.

Un moltiplicatore è configurato dall'utente e l'altro un moltiplicatore nascosto/non configurabile.

Combinare questi moltiplicatori con il buffer di base derivato per questa porta, in questo caso 1200 come nello scenario 1.

Algoritmo:

Rapporto coda corrente = rapporto buffer coda per la coda / classe da prevedere

Moltiplicatore nascosto = 400%

Moltiplicatore utente = Valore percentuale configurato in qos queue-softmax-multiplicator <percentuale>. Il valore predefinito è 100%

Softmax = ( Buffer base x (Rapporto coda corrente / 100)) x Moltiplicatore nascosto x (Moltiplicatore utente / 100)

<#root>

9500H(config)#

```
policy-map test2
```

```
9500H(config-pmap)#
```

```
class class1
```

```
9500H(config-pmap-c)#
```

```
priority level 1
```

```
9500H(config-pmap-c)#
```

```
queue-buffers ratio 50
```

```
<-- class 1 / first queue gets 50% of base buffer
```

```
9500H(config-pmap-c)#
```

```
class class-default
```

```
9500H(config-pmap-c)#
```

```
bandwidth remaining percent 100 <-- required configuration due to priority queue, can be ignored for thi
```

```
9500H(config-pmap-c)#
```

```
queue-buffers ratio 50
```

```
<-- class 2 / first queue gets 50% of base buffer
```

Riepilogo dei valori:

- Buffer di base = 1200 - Per trovare questo numero, vedere lo scenario 1
- Moltiplicatore utente = 100% - Valore predefinito, non ancora modificato
- Moltiplicatore nascosto = 4 - predefinito e non configurabile dall'utente
- Coda corrente (classe1) = 50% - Configurato nella mappa dei criteri con rapporto code-buffer 50
- Rapporto coda corrente (classe-default) = 50% - Configurato nella mappa dei criteri con un

rapporto di buffer della coda di 50

Determinare l'allocazione del buffer di classe 1:

Poiché class1 è una coda di priorità, riceve hardmax (buffer rigido) e un caso speciale di softmax non influenzato da moltiplicatori.

Classe1 hardmax = (Buffer base x Rapporto coda corrente(classe1) / 100)

Classe1 hardmax = 1200 x (50/100) = 600 - a causa di un caso speciale di una coda di priorità, arrestare tutti i calcoli matematici, assegnare il risultato a hardmax. Softmax equivale ad Hardmax come regola per il livello di priorità 1.

Determinare l'allocazione del buffer predefinito per classe:

Classe-default = (Buffer base x (Rapporto coda corrente (classe-default / 100)) x Moltiplicatore nascosto x (Moltiplicatore utente / 100)

Class class-default = [

[buffer di base] 1200 x [rapporto coda corrente] (50/100) = 600

[risultato precedente] 600 x [moltiplicatore nascosto] 4 x [moltiplicatore utente] (100/100) = 2400

]

<#root>

9500H(config)#

int tw1/0/3

9500H(config-if)#

service-policy output test2

<-- apply the policy

9500H#

```
show platform hardware fed active qos queue config interface twl/0/3
```

```
Asic:0 Core:1 DATA Port:22 GPN:103 LinkSpeed:0x12
AFD:Disabled FlatAFD:Disabled QoSMap:0 HW Queues: 176 - 183
  DrainFast:Disabled PortSoftStart:5 - 3600 BufferSharing:Disabled
  DTS Hardmax Softmax PortSMin GlbSMin PortStEnd QEnable
  ---- -
0 1 5
```

600

9

600

```
0 0 0 0 1 4800 En
```

<-- Hardmax is 600 as predicted, Softmax is set equal to Hardmax due to priority level 1

```
1 1 0 0 10
```

2400

```
16 1200 8 600 1 4800 En
```

<--

Softmax is 2400 as predicted

<snip>

Risultato finale: Q0 - Hardmax: 600 Softmax 600. Q1 - Softmax: 2400

Scenario 2.1: Modifica moltiplicatore configurato dall'utente dallo scenario 2

Questo scenario inizia in modo analogo allo scenario 2, con la differenza che ora è possibile configurare qos queue-softmax-multiplier 1200.

In questo modo i buffer softmax nella configurazione corrente vengono moltiplicati per 1200% o per un fattore di 12.

Riepilogo dei valori:

- Buffer di base = 1200 - Per trovare questo numero, vedere lo scenario 1
- Moltiplicatore utente = 1200% - Valore modificato in questo scenario, tramite qos queue-softmax-multiplier 1200
- Moltiplicatore nascosto = 4 - predefinito e non configurabile dall'utente
- Coda corrente (classe1) = 50% - Configurato nella mappa dei criteri con rapporto code-buffer 50
- Coda corrente (classe-default) = 50% - Configurato nella mappa dei criteri con rapporto code-buffer 50

Determinare l'allocazione del buffer di classe 1:

Poiché class1 è una coda di priorità, riceve hardmax (buffer rigido) e un caso speciale di softmax non influenzato da moltiplicatori.

Classe1 hardmax = (Buffer base x Rapporto coda corrente(classe1) / 100)

Classe1 hardmax =  $1200 \times (50/100) = 600$  - a causa di un caso speciale di una coda di priorità, arrestare tutti i calcoli matematici, assegnare il risultato a hardmax. Softmax equivale ad Hardmax come regola per il livello di priorità 1.

Determinare l'allocazione del buffer predefinito per classe:

Classe-default = (Buffer base x (Rapporto coda corrente (classe-default / 100)) x Moltiplicatore nascosto x (Moltiplicatore utente / 100)

Class class-default =[

[buffer di base] 1200 x [rapporto coda corrente] (50/100) = 600

[risultato precedente] 600 x [moltiplicatore nascosto] 4 x [moltiplicatore utente] (1200/100) = 28800

]

Configurare qos queue-softmax-multiplier 1200 e osservare le modifiche apportate a softmax (softmax è un valore massimo del buffer per la coda, ridimensionato in modo dinamico in base all'utilizzo complessivo corrente del buffer):

<#root>

9500H(config)#

```
qos queue-softmax-multiplier 1200
```

9500H#

```
show platform hardware fed active qos queue config interface tw1/0/3
```

```
Asic:0 Core:1 DATA Port:22 GPN:103 LinkSpeed:0x12
AFD:Disabled FlatAFD:Disabled QoSMap:0 HW Queues: 176 - 183
  DrainFast:Disabled PortSoftStart:3 - 31500 BufferSharing:Disabled
  DTS   Hardmax   Softmax   PortSMin   G1b1SMin   PortStEnd   QEnable
-----
0    1    5
```

600

5

600

```
0    0    0    0    6 42000    En
```

<-- Queue 0 does not change as its configured with priority level 1

```
1    1    0    0    6
```

28800

```
1    900    1    900    6 42000    En
```

<-- Softmax increases by 12x to 28800 from 1200 due to queue-softmax-multiplier 1200

<snip>

Risultato finale: Q1 - Hardmax: 600, Softmax 600. Q2 - Softmax: 28800

Scenario 3: Determinare il buffer soft e hard per più code e un rapporto implicito

In questo scenario, sono configurate cinque code, ma solo quattro hanno il rapporto dei buffer di

coda definito in modo esplicito. Il buffer allocato a queste code è lo stesso degli esempi precedenti.

La coda non configurata riceve la differenza tra la somma di tutti i buffer di coda configurati e 100.

Somma dei rapporti configurati in modo esplicito = (rapporto buffer Q0) + (rapporto buffer Q1) ... (rapporto buffer finale) - Sugli switch Catalyst serie 9000 sono supportate fino a 8 code, pertanto è possibile aggiungere fino a 8 rapporti

Rapporto implicito rimanente = (100 - Somma dei rapporti configurati in modo esplicito).

Rapporto implicito rimanenti è il valore assegnato a una coda per la quale non è configurato il rapporto dei buffer di coda.

Mapping dei criteri in uso per questo scenario:

```
<#root>
```

```
9500H(config)#
```

```
policy-map test3
```

```
9500H(config-pmap)#
```

```
class class1
```

```
9500H(config-pmap-c)#
```

```
priority level 1
```

```
9500H(config-pmap-c)#
```

```
queue-buffers ratio 20
```

```
9500H(config-pmap-c)#
```

```
class class2
```

```
9500H(config-pmap-c)#
```

```
bandwidth remaining percent 10 <-- no queue-buffers ratio statement for this class
```

```
9500H(config-pmap-c)#
```

```
class class3
```

```
9500H(config-pmap-c)#
```

```
bandwidth remaining percent 10
```

```
9500H(config-pmap-c)#
```

```
queue-buffers ratio 10
```

```
<-- rest of queues have an explicit queue-buffers ratio
```

```
9500H(config-pmap-c)#
```

```
class class4
```

```
9500H(config-pmap-c)#
```

```
bandwidth remaining percent 10
```

```
9500H(config-pmap-c)#
```

```
queue-buffers ratio 10
```

```
9500H(config-pmap-c)#
```

```
class class-default
```

```
9500H(config-pmap-c)#
```

```
bandwidth remaining percent 70
```

```
9500H(config-pmap-c)#
```

```
queue-buffers ratio 40
```

Riepilogo dei valori:

- Buffer di base = 1200 - Per trovare questo numero, vedere lo scenario 1
- Moltiplicatore utente = 100% - Valore predefinito, non ancora modificato
- Moltiplicatore nascosto = 4 - predefinito e non configurabile dall'utente
- Rapporto coda corrente (classe1) = 20% - Configurato nella mappa dei criteri con rapporto code-buffer 20
- Rapporto coda corrente(classe2) = Sconosciuto, è necessario risolvere per questo - rapporto coda-buffer non configurato per questa classe
- Coda corrente (classe3) = 10% - Configurato nella mappa dei criteri con rapporto code-buffer 10
- Rapporto coda corrente (classe4) = 10% - Configurato nella mappa dei criteri con rapporto buffer coda 10
- Coda corrente (classe-default) = 40% - Configurato nella mappa dei criteri con rapporto code-buffer 40

Calcola il rapporto buffer della coda che rimane:

Rapporto implicito rimanente = (100 - Somma dei rapporti configurati in modo esplicito).

$$100 - (20) - (10) - (10) - (50) = 20$$

Rapporto coda corrente(classe2) = 20

Calcola allocazione buffer coda finale

Classe1 = [Buffer base] 1200 x [Rapporto coda corrente(classe1)] (20/100) = 240 - Coda priorità, nessun ulteriore calcolo

Classe2 = [

[Buffer base] 1200 x [Rapporto coda corrente(classe2)] (10/100) = 240 - Allocazione buffer base per questa coda, ma deve essere moltiplicato per ottenere softmax per una coda non prioritaria  
[Allocazione buffer di base per questa coda] 120 x [Moltiplicatore nascosto] 4 x [Moltiplicatore utente] (100/100) = 960

]

Ripetere per le code rimaste:

Classe 3 = [

1200 x (10/100) = 120  
120 x 4 x (100/100) = 480  
]

Classe 4 = [

1200 x (10/100) = 120  
120 x 4 x (100/100) = 480  
]

Class class-default = [

1200 x 40/100 = 480  
600 x 4 x (100/100) = 1920  
]

Risultato della mappa dei criteri test3 applicata rispetto alla previsione:

<#root>

9500H(config)#

int tw1/0/3

9500H(config-if)#

service-policy output test3

9500H#

show platform hardware fed active qos queue config interface tw1/0/3

Asic:0 Core:1 DATA Port:22 GPN:103 LinkSpeed:0x12  
AFD:Disabled FlatAFD:Disabled QoSMap:0 HW Queues: 176 - 183  
DrainFast:Disabled PortSoftStart:4 - 2880 BufferSharing:Disabled  
DTS Hardmax Softmax PortSMin G1b1SMin PortStEnd QEnable  
-----  
0 1 5 240 8 240 0 0 0 0 6 3840 En  
1 1 0 0 9

960

16 480 8 240 6 3840 En

```
<-- queue without queue buffers ratio configured receives any leftover ratio, as predicted
```

```
2  1  0    0 11  480 16  240  8  120  6 3840    En
3  1  0    0 11  480 16  240  8  120  6 3840    En
4  1  0    0  4 1920 16  960  8  480  6 3840    En
<snip>
```

Risultato finale: Q0 - Hardmax: 240, Softmax 240. Q1 - Softmax: 960, Q2 - Softmax: 480, Q3 - Softmax: 480, Q4 - Softmax: 480

### Scenario 3.1: Determinazione del buffer soft e hard per più code e rapporti impliciti multipli

In questo scenario, vengono configurate cinque code e per due code non viene configurato il rapporto dei buffer delle code.

Per determinare l'allocazione del buffer, continua la stessa logica dello scenario 2, ma è necessario anche dividere il valore di Percentuale implicita rimanente per il numero totale di code implicite / code che non hanno il rapporto coda-buffer

Mapping dei criteri in uso per questo scenario:

```
<#root>
```

```
9500H(config)#
```

```
policy-map test4
```

```
9500H(config-pmap)#
```

```
class class1
```

```
9500H(config-pmap-c)#
```

```
priority level 1
```

```
9500H(config-pmap-c)#
```

```
queue-buffers ratio 20
```

```
9500H(config-pmap-c)#
```

```
class class2
```

```
9500H(config-pmap-c)#
```

```
bandwidth remaining percent 10 <-- no queue-buffers ratio statement for this class
```

```
9500H(config-pmap-c)#
```

```
class class3
```

```
9500H(config-pmap-c)#
```

```
bandwidth remaining percent 10 <-- no queue-buffers ratio statement for this class
```

```
9500H(config-pmap-c)#
```

```
class class4
```

```
9500H(config-pmap-c)#
```

```
bandwidth remaining percent 10
```

```
9500H(config-pmap-c)#
```

```
queue-buffers ratio 10
```

```
9500H(config-pmap-c)#
```

```
class class-default
```

```
9500H(config-pmap-c)#
```

```
bandwidth remaining percent 70
```

```
9500H(config-pmap-c)#
```

```
queue-buffers ratio 40
```

Riepilogo dei valori:

- Buffer di base = 1200 - Per trovare questo numero, vedere lo scenario 1
- Moltiplicatore utente = 100% - Valore predefinito

- Moltiplicatore nascosto = 4 - Predefinito e non configurabile dall'utente
- Rapporto coda corrente (classe1) = 20% - Configurato nella mappa dei criteri con rapporto code-buffer 20
- Rapporto coda corrente(classe2) = Sconosciuto, è necessario risolvere per questo - rapporto coda-buffer non configurato per questa classe
- Rapporto coda corrente(classe3) = Sconosciuto, è necessario risolvere per questo - rapporto coda-buffer non configurato per questa classe
- Rapporto coda corrente (classe4) = 10% - Configurato nella mappa dei criteri con rapporto buffer coda 10
- Rapporto coda corrente (classe-default) = 40% - Configurato nella mappa dei criteri con un rapporto di buffer della coda di 40

Calcola il rapporto buffer della coda che rimane:

Rapporto implicito rimanente =  $(100 - \text{Somma dei rapporti configurati in modo esplicito})$ .

Numero di code implicite = 2 (per la classe 2 e la classe 3 non è definito alcun rapporto coda-buffer)

Somma dei rapporti configurati =  $20+40+10 = 70$

Rapporto implicito rimasto =  $100 - 70 = 30$

Allocazione proporzioni coda implicita =  $[\text{Rapporto implicito rimanente}] 30 / [\text{Numero di code implicite}] 2 = 15$

Calcola allocazione buffer coda finale:

Classe1 =

$[\text{Buffer base}] 1200 \times [\text{Rapporto coda corrente}(\text{classe1})] (20/100) = 240$  - Coda priorità, nessun ulteriore calcolo

Classe2 =

$[\text{Buffer di base}] 1200 \times [\text{Allocazione proporzioni coda implicita}] (15/100) = 180$  - Poiché la classe 2 non ha una proporzione definita coda-buffer, il resto della proporzione coda-buffer dalle code esplicite viene condiviso tra le code implicite.

[Allocazione buffer di base per questa coda] 180 x [Moltiplicatore nascosto] 4 x [Moltiplicatore utente] (100/100) = 720

Ripetere per le code rimaste:

Classe 3 = [

1200 x 15/100 = 180  
120 x 4 x (100/100) = 720  
]

Classe 4 = [

1200 x (10/100) = 120  
120 x 4 x (100/100) = 480  
]

Class class-default = [

1200 x 40/100 = 480  
600 x 4 x (100/100) = 1920  
]

Risultato della mappa dei criteri test4 applicata rispetto alla previsione:

```
<#root>
```

```
9500H(config)#
```

```
interface tw1/0/3
```

```
9500H(config-if)#
```

```
service-policy output test4
```

```
9500H#
```

```
show platform hardware fed active qos queue config interface tw1/0/3
```

```
Asic:0 Core:1 DATA Port:22 GPN:103 LinkSpeed:0x12
```

```

AFD:Disabled FlatAFD:Disabled QoSMap:0 HW Queues: 176 - 183
  DrainFast:Disabled PortSoftStart:4 - 2880 BufferSharing:Disabled
  DTS Hardmax Softmax PortSMin GlbSMin PortStEnd QEnable
  -----
0  1  5  240  8  240  0  0  0  0  6  3840  En
1  1  0   0  9

```

720

```

16  360  8  180  6  3840  En

```

<-- queue 1 and 2

were not configured with queue-buffers ratio

```

2  1  0   0  9

```

720

```

16  360  8  180  6  3840  En

```

<-- queue 1 and 2 get an equal share of leftover buffer ratio

```

3  1  0   0  11  480  16  240  8  120  6  3840  En
4  1  0   0  4  1920  16  960  8  480  6  3840  En
<snip>

```



Nota: se il risultato dell'allocazione del rapporto di coda Implicito non è un numero intero, non è possibile ottenere una condivisione uguale. Il risultato viene arrotondato per eccesso per le code precedenti nella mappa dei criteri e per difetto per le code successive. La somma finale del rapporto di buffer della coda allocato rimane 100, ma le code implicite non sempre ottengono la stessa allocazione a causa del requisito del risultato integer appena descritto.

## Scenario 4: La somma del rapporto buffer coda tra le code è inferiore a 100

In questo scenario, vengono configurate cinque code, tutte con il rapporto code-buffer. La somma totale del rapporto tra coda e buffer nelle classi è inferiore a 100.

In questo caso, il rapporto del buffer non allocato viene distribuito uniformemente tra le classi.

Analogamente allo scenario precedente, se il risultato diviso del rapporto buffer coda rimasta non

è un numero intero, l'allocazione finale per ogni coda viene arrotondata per eccesso o per difetto e aggiunta al rapporto buffer coda configurato.

Mapping dei criteri in uso per questo scenario:

```
<#root>
```

```
9500H(config)#
```

```
policy-map test5
```

```
9500H(config-pmap)#
```

```
class class1
```

```
9500H(config-pmap-c)#
```

```
priority level 1
```

```
9500H(config-pmap-c)#
```

```
queue-buffers ratio 10
```

```
9500H(config-pmap-c)#
```

```
class class2
```

```
9500H(config-pmap-c)#
```

```
bandwidth remaining percent 10
```

```
9500H(config-pmap-c)#
```

```
queue-buffers ratio 10
```

```
9500H(config-pmap-c)#
```

```
class class3
```

```
9500H(config-pmap-c)#
```

```
bandwidth remaining percent 10
```

9500H(config-pmap-c)#

queue-buffers ratio 10

9500H(config-pmap-c)#

class class4

9500H(config-pmap-c)#

bandwidth remaining percent 10

9500H(config-pmap-c)#

queue-buffers ratio 10

9500H(config-pmap-c)#

class class-default

9500H(config-pmap-c)#

bandwidth remaining percent 70

9500H(config-pmap-c)#

queue-buffers ratio 12

Riepilogo dei valori:

- Buffer di base = 1200 - Per trovare questo numero, vedere lo scenario 1
- Moltiplicatore utente = 100% - Valore predefinito
- Moltiplicatore nascosto = 4 - Predefinito e non configurabile dall'utente
- Coda corrente (classe1) = 10% - Configurato nella mappa dei criteri con rapporto code-buffer 10
- Rapporto coda corrente (classe 2) = 10% - Configurato nella mappa dei criteri con rapporto buffer coda 10
- Rapporto coda corrente (classe3) = 10% - Configurato nella mappa dei criteri con rapporto buffer coda 10
- Rapporto coda corrente (classe4) = 10% - Configurato nella mappa dei criteri con rapporto buffer coda 10

- Coda corrente (classe-default) = 12% - Configurato nella mappa dei criteri con rapporto code-buffer 12
- Numero totale di code = 5

Somma dei rapporti configurati =  $10 + 10 + 10 + 10 + 10 + 12 = 52$

Rapporto buffer rimanente =  $100\% - 52\% = 48\%$

[Rapporto buffer rimanente] 48% / [Numero totale di code] 5 = 9,6% aggiunto per coda - Non si tratta di un numero intero, quindi l'applicazione finale alle code deve essere arrotondata per eccesso o per difetto per ogni coda

Per ottenere il numero di buffer della coda finale utilizzato dal sistema, è necessario aggiungere 9 o 10 al rapporto dei buffer della coda già configurato.

Alle classi superiori nella mappa dei criteri viene assegnato il valore arrotondato per eccesso, ovvero 10. Alle classi inferiori nella mappa dei criteri viene assegnato il valore arrotondato per difetto, ovvero 9.

Calcola allocazione buffer coda finale

Rapporto buffer residuo = 48

Class1 = [Buffer base] x ([Rapporto coda corrente(class1) + Valore arrotondato per eccesso del rapporto buffer condiviso rimasto])

Classe1 =  $1200 \times ((10\% + 10\%)/100) = 240$  - coda priorità, nessun ulteriore calcolo

Rapporto buffer rimanente =  $(48 - 10) = 38$

Class2 = [Buffer di base] x ([Rapporto coda corrente(class2) + Valore arrotondato per eccesso del rapporto buffer condiviso rimasto])

Classe2 =  $1200 \times ((10\% + 10\%)/100) = 240$  - Continuare a moltiplicare per i moltiplicatori di utente e di sistema poiché non si tratta di una coda di priorità

Class2 = [allocazione buffer di base per la coda] 240 x [Moltiplicatore nascosto] 4 x [Moltiplicatore utente]  $(100/100) = 960$  - risultato softmax per la coda

Rapporto buffer residuo =  $(38 - 10) - 28$

Ripetere per le code rimaste:

Classe 3 = [

$1200 \times ((10+10)/100) = 240$

$120 \times 4 \times (100/100) = 960$

]

Rapporto buffer rimanente =  $(28 - 10) = 18$

Classe 4 = [

$1200 \times ((10+9)/100) = 240$

$120 \times 4 \times (100/100) = 912$

]

Rapporto buffer rimanente = 9

Class class-default= [

$1200 \times ((12+9)/100) = 252$

$120 \times 4 \times (100/100) = 1008$

]

Rapporto buffer residuo = 0

Risultato della mappa dei criteri test5 applicata rispetto alla previsione:

<#root>

9500H#

```
show platform hardware fed active qos queue config interface tw1/0/3
```

```
Asic:0 Core:1 DATA Port:22 GPN:103 LinkSpeed:0x12
```

```
AFD:Disabled FlatAFD:Disabled QoSMap:0 HW Queues: 176 - 183
```

```
DrainFast:Disabled PortSoftStart:5 - 1512 BufferSharing:Disabled
```

```
DTS Hardmax Softmax PortSMin G1b1SMin PortStEnd QEnable
```

```
-----  
0 1 5 240 9
```

240

```
0 0 0 0 6 2016 En
1 1 0 0 10
```

960

```
16 480 8 240 6 2016 En
2 1 0 0 10
```

960

```
16 480 8 240 6 2016 En
3 1 0 0 11
```

912

```
16 456 8 228 6 2016 En
4 1 0 0 12
```

1008

```
16 504 8 252 6 2016 En
<snip>
```

## Scenario 5: Determinazione del soft buffer e del buffer di una classe con priorità di livello 2

In questo scenario, una classe in una mappa dei criteri viene configurata con il livello di priorità 2.

A differenza del livello di priorità 1, in cui softmax non è influenzato dai moltiplicatori ed è impostato su hardmax, il livello di priorità 2 consente di moltiplicare softmax mentre ha anche un'allocazione hard buffer (hardmax).

Mapping dei criteri in uso per questo scenario:

```
<#root>
```

```
9500H(config)#
```

```
policy-map test6
```

```
9500H(config-pmap)#
```

```
class class1
```

```
9500H(config-pmap-c)#
```

```
priority level 1
```

```
9500H(config-pmap-c)#
```

```
queue-buffers ratio 50 <-- 50 / 50 split between both queues
```

```
9500H(config-pmap-c)#
```

```
class class-default
```

```
9500H(config-pmap-c)#
```

```
priority level 2
```

```
<-- Priority level 2 in use now
```

```
9500H(config-pmap-c)#
```

```
queue-buffers ratio 50 <-- 50 / 50 split between both queues
```

Risultato della mappa dei criteri test6 applicata:

```
<#root>
```

```
9500H#
```

```
show platform hardware fed active qos queue config interface tw1/0/3
```

```
Asic:0 Core:1 DATA Port:22 GPN:103 LinkSpeed:0x12
AFD:Disabled FlatAFD:Disabled QoSMap:0 HW Queues: 176 - 183
  DrainFast:Disabled PortSoftStart:5 - 3600 BufferSharing:Disabled
    DTS  Hardmax  Softmax  PortSMin  Glb1SMin  PortStEnd  QEnable
  -----
0  1  5  600  9
```

```
600
```

```
0  0  0  0  1  4800  En
```

```
<-- Softmax is equal to hardmax
```

```
1 1 5 600 10
```

```
2400
```

```
16 1200 0 0 1 4800 En
```

```
<-- Softmax is multiplied by Hidden Multiplier (400%) and User Multiplier (100% default)r
```

```
<snip>
```

Nell'output mostrato in precedenza, la seconda coda softmax ha 4 softmax della prima coda. Ciò è dovuto al fatto che il softmax del livello di priorità 1 non è influenzato in modo specifico dai moltiplicatori softmax del sistema, mentre il livello di priorità 2 è.

Se si configura un moltiplicatore softmax dell'utente, viene interessata solo la coda di livello di priorità 2:

```
<#root>
```

```
9500H(config)#
```

```
qos queue-softmax-multiplier 200
```

```
<#root>
```

```
9500H#
```

```
show platform hardware fed active qos queue config interface tw1/0/3
```

```
Asic:0 Core:1 DATA Port:22 GPN:103 LinkSpeed:0x12
AFD:Disabled FlatAFD:Disabled QoSMap:0 HW Queues: 176 - 183
  DrainFast:Disabled PortSoftStart:5 - 7200 BufferSharing:Disabled
  DTS Hardmax Softmax PortSMin GblSMin PortStEnd QEnable
  -----
0 1 5 600 9 600 0 0 0 0 5 9600 En
```

```
<--- priority-level 1 queue unaffected by softmax multiplier
```

```
1 1 5 600 10
```

```
4800
```

```
8 1200 0 0 5 9600 En
```

```
<--- User multiplier increased to 200%, softmax for this queue doubles
```

```
<snip>
```

## Limite coda

La configurazione del limite di coda influisce sull'allocazione finale del buffer della coda

Il meccanismo principale per influenzare l'allocazione del buffer della coda è la configurazione del rapporto buffer della coda aggiunta per coda in una mappa dei criteri MQC.

Tuttavia, l'allocazione del buffer della coda è influenzata da altre configurazioni.

Queue-limit definisce le soglie in base alle quali viene scartata una particolare classe di traffico (tramite Weighted Tail Drop, WTD), che non sono descritte nel presente documento.

In una circostanza specifica, queue-limit modifica il moltiplicatore nascosto di sistema per le code di buffer soft, che influisce sull'allocazione complessiva del buffer soft per la coda a cui viene applicato il limite di coda.

Innanzitutto, è necessario tenere presente che il limite di coda può essere configurato fino a 3 volte per classe. Questo definisce fino a 3 soglie per WTD su base DSCP o CoS.

Nell'output successivo, vengono definite solo due soglie.

Applicare due limiti di coda a una mappa dei criteri:

```
<#root>
```

```
9500H(config)#
```

```
policy-map test7
```

```
9500H(config-pmap)#
```

```
class class1
```

```
9500H(config-pmap-c)#
```

```
priority level 1
```

```
9500H(config-pmap-c)#
```

```
queue-buffers ratio 50
```

```
9500H(config-pmap-c)#
```

```
class class-default
```

```
9500H(config-pmap-c)#
```

```
priority level 2
```

```
9500H(config-pmap-c)#
```

```
queue-buffers ratio 50
```

```
9500H(config-pmap-c)#
```

```
queue-limit dscp af11 percent 10 <-- Tells system to drop af11 traffic at 10% queue utilization
```

```
9500H(config-pmap-c)#
```

```
queue-limit dscp af12 percent 50 <-- Tells system to drop af12 traffic at 50% queue utilization
```

Osservare i risultati dell'allocazione dei buffer:

```
<#root>
```

```
9500H(config-pmap-c)#
```

```
interface tw1/0/3
```

```
9500H(config-if)#
```

```
service-policy output test7
```

```
9500H#
```

```
show platform hardware fed active qos queue config interface tw1/0/3
```

```
Asic:0 Core:1 DATA Port:22 GPN:103 LinkSpeed:0x12
AFD:Disabled FlatAFD:Disabled QoSMap:0 HW Queues: 176 - 183
  DrainFast:Disabled PortSoftStart:5 - 7200 BufferSharing:Disabled
  DTS Hardmax Softmax PortSMin GblSMin PortStEnd QEnable
  ---- -
0  1  5  600  9  600  0  0  0  0  5  9600  En
1  1  5  600 10
```

```
4800
```

```
8 1200 0 0 5 9600 En
```

```
<--- final result for queue that contains 2 queue-limit statements is 4800
```

```
<snip>
```

Nell'esempio successivo, una terza configurazione del limite di coda viene aggiunta a class-default.

Osservare i risultati dell'allocazione dei buffer:

```
<#root>
```

```
9500H(config)#
```

```
policy-map test7
```

```
9500H(config-pmap)#
```

```
class class-default
```

```
9500H(config-pmap-c)#
```

```
queue-limit dscp af13 percent 100
```

```
9500H#
```

```
show platform hardware fed active qos queue config interface tw1/0/3
```

```

Asic:0 Core:1 DATA Port:22 GPN:103 LinkSpeed:0x12
AFD:Disabled FlatAFD:Disabled QoSMap:0 HW Queues: 176 - 183
  DrainFast:Disabled PortSoftStart:5 - 1800 BufferSharing:Disabled
    DTS  Hardmax  Softmax  PortSMin  Glb1SMin  PortStEnd  QEnable
  -----
0  1  5  600  9  600  0  0  0  0  5  2400  En
1  1  5  600  10

```

```
1200
```

```
32 1200 0 0 5 2400 En
```

```
<-- Softmax reduces by 400% from previous example
```

```
<snip>
```

Quando si aggiunge una terza configurazione per il limite di coda a una coda, il moltiplicatore nascosto del soft buffer del sistema (400%) viene disattivato per quella coda. Tuttavia, tale coda rispetta ancora un utente configurato qos queue-softmax-multiplier <percentuale>.

## Informazioni correlate

- [Piattaforme di switching Cisco Catalyst 9000: White paper su QoS e code](#)
- [Supporto tecnico Cisco e download](#)

## Informazioni su questa traduzione

Cisco ha tradotto questo documento utilizzando una combinazione di tecnologie automatiche e umane per offrire ai nostri utenti in tutto il mondo contenuti di supporto nella propria lingua. Si noti che anche la migliore traduzione automatica non sarà mai accurata come quella fornita da un traduttore professionista. Cisco Systems, Inc. non si assume alcuna responsabilità per l'accuratezza di queste traduzioni e consiglia di consultare sempre il documento originale in inglese (disponibile al link fornito).