

Chapter 5

Configuring Forwarding Classes

To configure class-of-service (CoS) forwarding classes, you can include the following statements at the [edit class-of-service] hierarchy level of the configuration:

```
class-of-service {
  forwarding-classes {
    queue queue-number class-name priority (high | low);
  }
  interfaces {
    interface-name {
      unit logical-unit-number {
        forwarding-class class-name;
      }
    }
  }
  restricted-queues {
    forwarding-class class-name queue queue-number;
  }
}
```

This chapter discusses the following topics:

- Configuring Forwarding Classes on page 34
- Default Forwarding Classes on page 34
- Assigning a Forwarding Class to an Interface on page 36
- Overriding Fabric Priority Queuing on page 36
- Configuring Up to Eight Forwarding Classes on page 36

Configuring Forwarding Classes

You assign each forwarding class to an internal queue number by including the `forwarding-classes` statement at the `[edit class-of-service]` hierarchy level:

```
[edit class-of-service]
forwarding-classes {
  queue queue-number class-name;
}
```



NOTE: You cannot commit a configuration that assigns the same forwarding class to two different queues.

Default Forwarding Classes

Table 9 shows the four forwarding classes defined by default.

Table 9: Default Forwarding Classes

Forwarding Class Name	Queue
best-effort	queue 0
expedited-forwarding	queue 1
assured-forwarding	queue 2
network-control	queue 3

The following rules govern queue assignment:

- If classifiers fail to classify a packet, the packet always receives the default classification to the class associated with queue 0.
- The number of queues is dependent on the hardware plugged into the chassis. CoS configurations are inherently contingent on the number of queues on the system. Only two classes, `best-effort` and `network-control`, are actually referenced in the default configuration. The default configuration works on any platform.
- CoS configurations that specify more queues than the platform can support are not accepted. The commit fails with a detailed message that states the total number of queues available.

- All default CoS configuration is based on queue number. The name of the forwarding class that shows up when the default configuration is displayed is the forwarding class currently associated with that queue.

This is the default configuration for **forwarding-classes**:

```
[edit class-of-service]
forwarding-classes {
  queue 0 best-effort;
  queue 1 expedited-forwarding;
  queue 2 assured-forwarding;
  queue 3 network-control;
}
```

If you reassign the forwarding-class names, the **best-effort** forwarding-class name appears in the locations in the configuration previously occupied by **network-control** as follows:

```
forwarding-classes {
  queue 0 network-control;
  queue 1 assured-forwarding;
  queue 2 expedited-forwarding;
  queue 3 best-effort;
}
```

All the default rules of classification and scheduling that applied to queue 3 still apply. Queue 3 is simply now renamed **best-effort**.

- In the current default configuration:
 - Only IP precedence classifiers are associated with interfaces.
 - The only classes designated are **best-effort** and **network-control**.
 - Schedulers are not defined for the **expedited-forwarding** or **assured-forwarding** classes.
- You must make a conscious effort to classify packets to the **expedited-forwarding** or **assured-forwarding** class and define schedulers for these classes.
- For ATM interfaces on M-series platforms, when you use fixed classification with multiple logical interfaces classifying to separate queues, a logical interface without a classifier attached inherits the most recent classifier applied on a different logical interface. For example, suppose you configure traffic through logical unit 0 to be classified into Q1, and you configure traffic through logical unit 1 to be classified into Q3. You want traffic through logical unit 2 to be classified into the default classifier, which is Q0. In this case, traffic through logical unit 2 is classified into Q3, because the configuration of logical unit 1 was committed last.

Assigning a Forwarding Class to an Interface

To assign the forwarding class configuration to the output logical interface, include the `forwarding-class` statement at the [edit class-of-service interfaces *interface-name* unit *logical-unit-number*] hierarchy level:

```
[edit class-of-service interfaces interface-name unit logical-unit-number]
forwarding-class class-name;
```

You can include interface wildcards for *interface-name* and *logical-unit-number*.

Overriding Fabric Priority Queuing

For M320 and T-series platforms only, you can override automatic fabric priority queuing. For egress interfaces, fabric priority queuing matches the queue priority you assign at the [edit class-of-service schedulers *scheduler-name*] hierarchy level. High-priority egress traffic is automatically assigned to high-priority fabric queues. Likewise, low-priority egress traffic is automatically assigned to low-priority fabric queues.

You can override the default fabric priority queuing of egress traffic by including the `priority` statement at the [edit class-of-service forwarding-classes queue *queue-number* *class-name*] hierarchy level:

```
[edit class-of-service forwarding-classes queue queue-number class-name]
priority (high | low);
```

For information about associating a scheduler with a fabric priority, see “Associating a Scheduler with a Fabric Priority” on page 82.

Configuring Up to Eight Forwarding Classes

By default on M-series and T-series platforms, four queues are mapped to four forwarding classes, as shown in Table 9 on page 34. For M320 and T-series platforms only, you can configure more than four forwarding classes and queues. To do this, include the `queue` statement at the [edit class-of-service forwarding-classes] hierarchy level:

```
[edit class-of-service forwarding-classes]
queue queue-number class-name;
```

For all M320 and T-series PICs, you can configure up to eight forwarding classes. For most T-series PICs, the output queue number can be from 0 through 7, and you must map the forwarding classes one-to-one with the output queues. This is the global configuration.

By default, the scheduler transmission rate and buffer size percentages for queues 0 through 7 are 95, 0, 0, 5, 0, 0, 0, and 0 percent.

For more detail, see the following sections:

- Enabling Eight Queues on Interfaces on page 37
- PICs Restricted to Four Queues on page 37
- Examples: Configuring up to Eight Forwarding Classes on page 39

For information about configuring eight forwarding classes on ATM2 IQ interfaces, see “Enabling Eight Queues on ATM2 IQ Interfaces” on page 137.

Enabling Eight Queues on Interfaces

By default, PICs on T-series and M320 platforms are restricted to a maximum of four egress queues per interface. You can enable eight egress queues on some interfaces by including the `max-queues-per-interface` statement at the `[edit chassis fpc slot-number pic pic-number]` hierarchy level:

```
[edit chassis fpc slot-number pic pic-number]
max-queues-per-interface (4 | 8);
```

The numerical value can be 4 or 8.

To determine how many queues an interface supports, you can check the CoS queues output field of the `show interfaces interface-name extensive` command:

```
user@host> show interfaces so-1/0/0 extensive
CoS queues: 8 supported
```

If you include the `max-queues-per-interface` statement, all ports on the PIC use the configured maximum. When you change between four queues and eight queues, all physical interfaces on the PIC are deleted and re-added.

For 4-port OC3c/STM1 PICs on M320 and T-series platforms, when you include the `max-queues-per-interface 8` statement, you can configure up to eight queues on ports 0 and 2. After you commit the configuration, the PIC goes offline and comes back online with only ports 0 and 2 operational. No interfaces can be configured on ports 1 and 3. If you do not include the `max-queues-per-interface` statement or if you include the `max-queues-per-interface 4` statement, you can use all four ports and configure up to four queues per port.

For more information, see the *JUNOS System Basics Configuration Guide*.

PICs Restricted to Four Queues

Some T-series PICs support up to eight forwarding classes and are restricted to four queues. For these PICs only, you can assign multiple forwarding classes to single queues. Contact Juniper Networks customer support for a current list of T-series PICs that are restricted to four queues. To determine how many queues an interface supports, you can check the CoS queues output field of the `show interfaces interface-name extensive` command:

```
user@host> show interfaces so-1/0/0 extensive
CoS queues: 8 supported
```

By default, for T-series PICs that are restricted to four queues, the routing platform overrides the global configuration based on the following formula:

$$Q_r = Q_d \text{ mod } R_{\text{max}}$$

Q_r is the queue number assigned if the PIC is restricted to four queues.

Q_d is the queue number that would have been mapped if this PIC were not restricted.

R_{max} is the maximum number of restricted queues available. Currently, this is four.

For example, assume you map the forwarding class `ef` to queue 6. For a PIC restricted to four queues, the queue number for forwarding class `ef` is $Q_r = 6 \text{ mod } 4 = 2$.

To determine which queue is assigned to a forwarding class, use the `show class-of-service forwarding-class` command from the top level of the CLI. The output shows queue assignments for both global queue mappings and restricted queue mappings:

```
user@host> show class-of-service forwarding-class
Forwarding class      Queue  Restricted Queue  Fabric
priority
  be                   0      2                 low
  ef                   1      2                 low
  assured-forwarding  2      2                 low
  network-control     3      3                 low
```

For T-series PICs restricted to four queues, you can override the formula-derived queue assignment by including the `restricted-queues` statement at the `[edit class-of-service]` hierarchy level:

```
[edit class-of-service]
restricted-queues {
  forwarding-class class-name queue queue-number;
}
```

You can configure up to eight forwarding classes. The output queue number can be from 0 through 3. Therefore, for PICs restricted to four queues, you can map multiple forwarding classes to single queues. If you map multiple forwarding classes to a queue, the multiple forwarding classes must refer to the same scheduler. The class name you configure at the `[edit class-of-service restricted-queues]` hierarchy level must be either a default forwarding class name from Table 9 on page 34 or a forwarding class you configure at the `[edit class-of-service forwarding-classes]` hierarchy level.

Examples: Configuring up to Eight Forwarding Classes

Configure a one-to-one mapping between eight forwarding classes and eight queues.

For PICs restricted to four queues, map two forwarding classes to each queue.

```
[edit class-of-service]
forwarding-classes {
  queue 0 be;
  queue 1 ef;
  queue 2 af;
  queue 3 nc;
  queue 4 ef1;
  queue 5 ef2;
  queue 6 af1;
  queue 7 nc1;
}
```

Mapping Two Forwarding Classes to Each Queue

```
[edit class-of-service]
restricted-queues {
  forwarding-class be queue 0;
  forwarding-class be1 queue 0;
  forwarding-class ef queue 1;
  forwarding-class ef1 queue 1;
  forwarding-class af queue 2;
  forwarding-class af1 queue 2;
  forwarding-class nc queue 3;
  forwarding-class nc1 queue 3;
}
```

Defining Eight Classifiers

```
[edit class-of-service]
classifiers {
  dscp dscp-table {
    forwarding-class ef {
      loss-priority low code-points [101000, 101001];
      loss-priority high code-points [101010, 101011];
    }
    forwarding-class af {
      loss-priority low code-points [010000, 010001];
      loss-priority high code-points [010010, 010011];
    }
    forwarding-class be {
      loss-priority low code-points [000000];
    }
    forwarding-class nc {
      loss-priority low code-points [111000];
    }
    forwarding-class ef1 {
      loss-priority low code-points [101100, 101101];
      loss-priority high code-points [101110];
    }
    forwarding-class af1 {
      loss-priority high code-points [101110];
    }
  }
}
```

```

        forwarding-class ef2 {
            loss-priority low code-points [1011111];
        }
        forwarding-class af2 {
            loss-priority low code-points [010000];
        }
    }
}

```

Adding Eight Schedulers to a Scheduler Map

Configure a custom scheduler map that applies globally to all interfaces, except those that are restricted to four queues.

```

[edit class-of-service]
scheduler-maps {
    interface-global {
        forwarding-class be scheduler Q0;
        forwarding-class ef scheduler Q1;
        forwarding-class af scheduler Q2;
        forwarding-class nc scheduler Q3;
        forwarding-class ef1 scheduler Q4;
        forwarding-class ef2 scheduler Q5;
        forwarding-class af1 scheduler Q6;
        forwarding-class nc1 scheduler Q7;
    }
}
schedulers {
    Q0 {
        transmit-rate percent 25;
        buffer-size percent 25;
        priority low;
        drop-profile-map loss-priority any protocol both drop-default;
    }
    Q1 {
        buffer-size temporal 2000;
        priority strict-high;
        drop-profile-map loss-priority any protocol both drop-ef;
    }
    Q2 {
        transmit-rate percent 35;
        buffer-size percent 35;
        priority low;
        drop-profile-map loss-priority any protocol both drop-default;
    }
    Q3 {
        transmit-rate percent 5;
        buffer-size percent 5;
        drop-profile-map loss-priority any protocol both drop-default;
    }
    Q4 {
        transmit-rate percent 5;
        priority high;
        drop-profile-map loss-priority any protocol both drop-ef;
    }
}

```

```

Q5 {
    transmit-rate percent 10;
    priority high;
    drop-profile-map loss-priority any protocol both drop-ef;
}
Q6 {
    transmit-rate remainder;
    priority low;
    drop-profile-map loss-priority any protocol both drop-default;
}
Q7 {
    transmit-rate percent 5;
    priority high;
    drop-profile-map loss-priority any protocol both drop-default;
}
}

```

Configuring a Scheduler Map Applicable to an Interface Restricted to Four Queues

For PICs restricted to four queues, if you map multiple forwarding classes to a queue, the multiple forwarding classes must refer to the same scheduler.

```

[edit class-of-service]
scheduler-maps {
    interface-restricted {
        forwarding-class be scheduler Q0;
        forwarding-class ef scheduler Q1;
        forwarding-class ef1 scheduler Q1;
        forwarding-class ef2 scheduler Q1;
        forwarding-class af1 scheduler Q2;
        forwarding-class af scheduler Q2;
        forwarding-class nc scheduler Q3;
        forwarding-class nc1 scheduler Q3;
    }
}
[edit class-of-service]
restricted-queues {
    forwarding-class be queue 0;
    forwarding-class ef queue 1;
    forwarding-class ef1 queue 1;
    forwarding-class ef2 queue 1;
    forwarding-class af queue 2;
    forwarding-class af1 queue 2;
    forwarding-class nc queue 3;
    forwarding-class nc1 queue 3;
}

```

**Configuring an IP
Precedence Classifier
and Rewrite Tables**

```
[edit class-of-service]
classifiers {
  inet-precedence inet-classifier {
    forwarding-class be {
      loss-priority low code-points 000;
    }
    forwarding-class af11 {
      loss-priority high code-points 001;
    }
    forwarding-class ef {
      loss-priority low code-points 010;
    }
    forwarding-class nc1 {
      loss-priority high code-points 011;
    }
    forwarding-class {
      loss-priority low code-points 100;
    }
    forwarding-class af12 {
      loss-priority high code-points 101;
    }
    forwarding-class ef1 {
      loss-priority low code-points 110;
    }
    forwarding-class nc2 {
      loss-priority high code-points 111;
    }
  }
}
exp exp-rw-table {
  forwarding-class be {
    loss-priority low code-point 000;
  }
  forwarding-class af11 {
    loss-priority high code-point 001;
  }
  forwarding-class ef {
    loss-priority low code-point 010;
  }
  forwarding-class nc1 {
    loss-priority high code-point 111;
  }
  forwarding-class be1 {
    loss-priority low code-point 100;
  }
  forwarding-class af12 {
    loss-priority high code-point 101;
  }
  forwarding-class ef1 {
    loss-priority low code-point 110;
  }
  forwarding-class nc2 {
    loss-priority low code-point 111;
  }
}
```

```
inet-precedence inet-rw-table {
  forwarding-class be {
    loss-priority low code-point 000;
  }
  forwarding-class af11 {
    loss-priority high code-point 001;
  }
  forwarding-class ef1 {
    loss-priority low code-point 010;
  }
  forwarding-class nc1 {
    loss-priority low code-point 111;
  }
  forwarding-class be1 {
    loss-priority low code-point 100;
  }
  forwarding-class af12 {
    loss-priority high code-point 101;
  }
  forwarding-class ef1 {
    loss-priority low code-point 111;
  }
  forwarding-class nc2 {
    loss-priority low code-point 110;
  }
}
```

