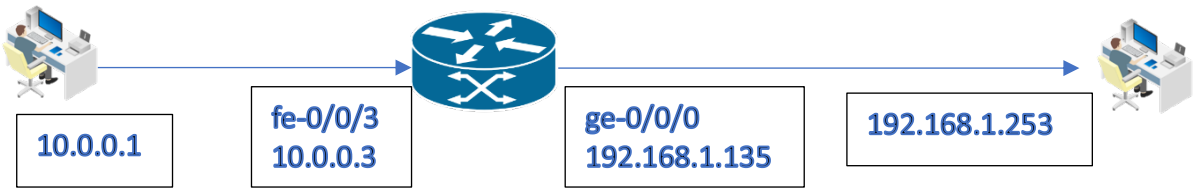
An Example of CoS classification- and marking with Junos



Preface

Client 10.0.0.1 will be routed through an SRX 210 in packet – mode.
The SRX will rewrite the DSCP value from 0 to best-effort 46.

Configuration on the SRX with a multifield classifier

Th ingress interface where PING from 10.0.0.1 comes in, is equipped with a multifield classifier that will classify this traffic for the **best-effort** standard forwarding-class.

```
labuser# show firewall
family inet {
    filter classify-traffic10 {
        term admin {
            from {
                source-address {
                    10.0.0.0/24;
                }
            }
            then {
                forwarding-class expedited-forwarding;
                accept;
            }
        }
    }
}
```

Associate the firewall filter / multifield classifier to the **Ingress** interface fe-0/0/3:

```
labuser# show interfaces fe-0/0/3
unit 0 {
    family inet {
        filter {
            input classify-traffic10;
        }
        address 10.0.0.35/24;
    }
}
```

For the rewriting of the unmarked traffic, a standard rewrite rule will be used on **Egress** interface ge-0/0/0:

```
labuser# show class-of-service
interfaces {
    ge-0/0/0 {
        unit 0 {
            rewrite-rules {
                dscp default;
```

```
            }
        }
    }
}
```

What this does is:

```
labuser# run show class-of-service rewrite-rule type dscp
Rewrite rule: dscp-default, Code point type: dscp, Index: 31
  Forwarding class                Loss priority       Code point
  best-effort                     low                 000000
  best-effort                     high                000000
  expedited-forwarding            low                 101110
  expedited-forwarding            high                101110
  assured-forwarding              low                 001010
  assured-forwarding              high                001100
  network-control                 low                 110000
  network-control                 high                111000
```

So IF traffic is classified as expedited-forwarding, (here due to the multifield classifier) it will be rewritten with DSCP : 101 110 = **46**. (voip)

Here a ping is received from 10.0.0.1 as it is forwarded through the SRX:

```
> Frame 1: 98 bytes on wire (784 bits), 98 bytes captured (784 bits) on interface
> Ethernet II, Src: JuniperN_1b:9b:c0 (00:26:88:1b:9b:c0), Dst: Apple_39:c6:f3 (1
v Internet Protocol Version 4, Src: 10.0.0.1, Dst: 192.168.1.253
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
  v Differentiated Services Field: 0xb8 (DSCP: EF PHB, ECN: Not-ECT)
      1011 10.. = Differentiated Services Codepoint: Expedited Forwarding (46)
      .... ..00 = Explicit Congestion Notification: Not ECN-Capable Transport (0)
```

The DSCP field is clearly rewritten to 46.

Classification using BA

In case packets come in from a source that already HAS marked the packets for CoS, classification can simply be performed using the “Behavior Aggregate” method where the above rewrite rule works in reverse, now as **classifier** on Ingress interface fe-0/0/3..

```
labuser# show class-of-service interfaces
fe-0/0/3 {
    unit 0 {
        classifiers {
            dscp default;
        }
    }
}
```

The previous multifield classifier association will be removed from the interface.

The proof:

Since the client is sending regular PING datagrams, (DSCP = 0) they will initially NOT be classified as other traffic then **best-effort**:

```
labuser# run show interfaces queue ge-0/0/0 | find best-effort
Physical interface: ge-0/0/0, Enabled, Physical link is Up
  Interface index: 134, SNMP ifIndex: 507
Forwarding classes: 8 supported, 4 in use
Egress queues: 8 supported, 4 in use
Queue: 0, Forwarding classes: best-effort
  Queued:
    Packets              :                     55                    1 pps
    Bytes                :                  11142                 1416 bps
```

Below you can see how ICMP traffic is received with a DSCP value of 0.

```
> Frame 1: 98 bytes on wire (784 bits), 98 bytes captured (784 bits) on interface en0, id 0
> Ethernet II, Src: JuniperN_1b:9b:c0 (00:26:88:1b:9b:c0), Dst: Apple_39:c6:f3 (14:98:77:39:c6:f3)
v Internet Protocol Version 4, Src: 10.0.0.1, Dst: 192.168.1.253
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
  v Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
      0000 00.. = Differentiated Services Codepoint: Default (0)
      .... ..00 = Explicit Congestion Notification: Not ECN-Capable Transport (0)
```

Using different DSCP values

So now we will use different DSCP values on the sending client, in order to get classified into different forwarding-classes / queues.

note: DSCP 46 will result in a TOS field of: % 101 110 00 = **184**.

First we clear the statistics in Junos:

```
labuser# run clear interfaces statistics all
```

So on the client (Linux MINT), the ping (count 5) looks as follows:

```
$ ping -Q 184   192.168.1.253. -c 5
```

Once again, the traffic is marked as **expedited-forwarding** and received with the marking DSCP 46:

```
> Frame 4: 98 bytes on wire (784 bits), 98 bytes captured (784 bits) on interface en0, id 0
> Ethernet II, Src: JuniperN_1b:9b:c0 (00:26:88:1b:9b:c0), Dst: Apple_39:c6:f3 (14:98:77:39:c6:f3)
v Internet Protocol Version 4, Src: 10.0.0.1, Dst: 192.168.1.253
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
  v Differentiated Services Field: 0xb8 (DSCP: EF PHB, ECN: Not-ECT)
      1011 10.. = Differentiated Services Codepoint: Expedited Forwarding (46)
      .... ..00 = Explicit Congestion Notification: Not ECN-Capable Transport (0)
```

The reason for this is the rewrite rule “dscp-default”:

```
labuser# run show class-of-service rewrite-rule type dscp
Rewrite rule: dscp-default, Code point type: dscp, Index: 31
  Forwarding class                Loss priority       Code point
  best-effort                     low                 000000
  best-effort                     high                000000
  expedited-forwarding            low                 101110
  expedited-forwarding            high                101110
```

```
  assured-forwarding             low                 001010
  assured-forwarding             high                001100
  network-control                low                 110000
  network-control                high                111000
```

And in the queues:

```
labuser# run show interfaces queue ge-0/0/0 | find expedited
Queue: 1, Forwarding classes: expedited-forwarding
  Queued:
    Packets              :                      5                0 pps
    Bytes                :                    490                0 bps
```

Let's try another value.. according to the rewrite rule "dscp", a DSCP value of 001 100 should result in **assured-forwarding**. For DSCP this is **12**.

In TOS this is: % 001 100 00 = 48.

```
$ ping -Q 48   192.168.1.253. -c 5
```

The queues:

```
labuser# run show interfaces queue ge-0/0/0 | find assured
Queue: 2, Forwarding classes: assured-forwarding
  Queued:
    Packets              :                      5                0 pps
    Bytes                :                    490                0 bps
```

On the receiving end:

```
Frame 67: 98 bytes on wire (784 bits), 98 bytes captured (784 bits) on interface en0, id 0
Ethernet II, Src: JuniperN_1b:9b:c0 (00:26:88:1b:9b:c0), Dst: Apple_39:c6:f3 (14:98:77:39:c6:f3)
Internet Protocol Version 4, Src: 10.0.0.1, Dst: 192.168.1.253
  0100 .... = Version: 4
  .... 0101 = Header Length: 20 bytes (5)
  Differentiated Services Field: 0x30 (DSCP: AF12, ECN: Not-ECT)
  0011 00.. = Differentiated Services Codepoint: Assured Forwarding 12 (12)
  .... ..00 = Explicit Congestion Notification: Not ECN-Capable Transport (0)
```

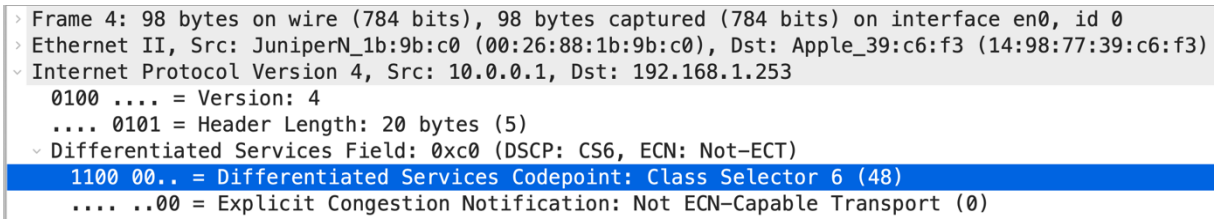
The last test will use the value of DSCP value of % 110 000 = 48 = TOS % 110 000 00 = 192.

Note: this happens to be the value used by OSPF. Hence OSPF traffic is generally classified as "network-control" traffic.

```
$ ping -Q 192   192.168.1.253. -c 5
```

The queues:

```
labuser# run show interfaces queue ge-0/0/0 | find network-control
Queue: 3, Forwarding classes: network-control
  Queued:
    Packets              :                      5                0 pps
    Bytes                :                    664                0 bps
```



Note: there is also a DSCP filter in wireshark: ip.dsfield.dscp == 48

What remains about CoS

Is a lot. This document only describes classification with BA and multifield classifiers. Another methodology is classification using a Policer. Or using fixed classification where simply “all traffic incoming on this interfaces is classified into forwarding-class ….”.

And of course, there is the configuration of schedulers and scheduler-maps. And then there is shaping (not that useful for real time – applications if you ask me) and drop-profiles.
But those techniques are out of scope for todays’ discussion.