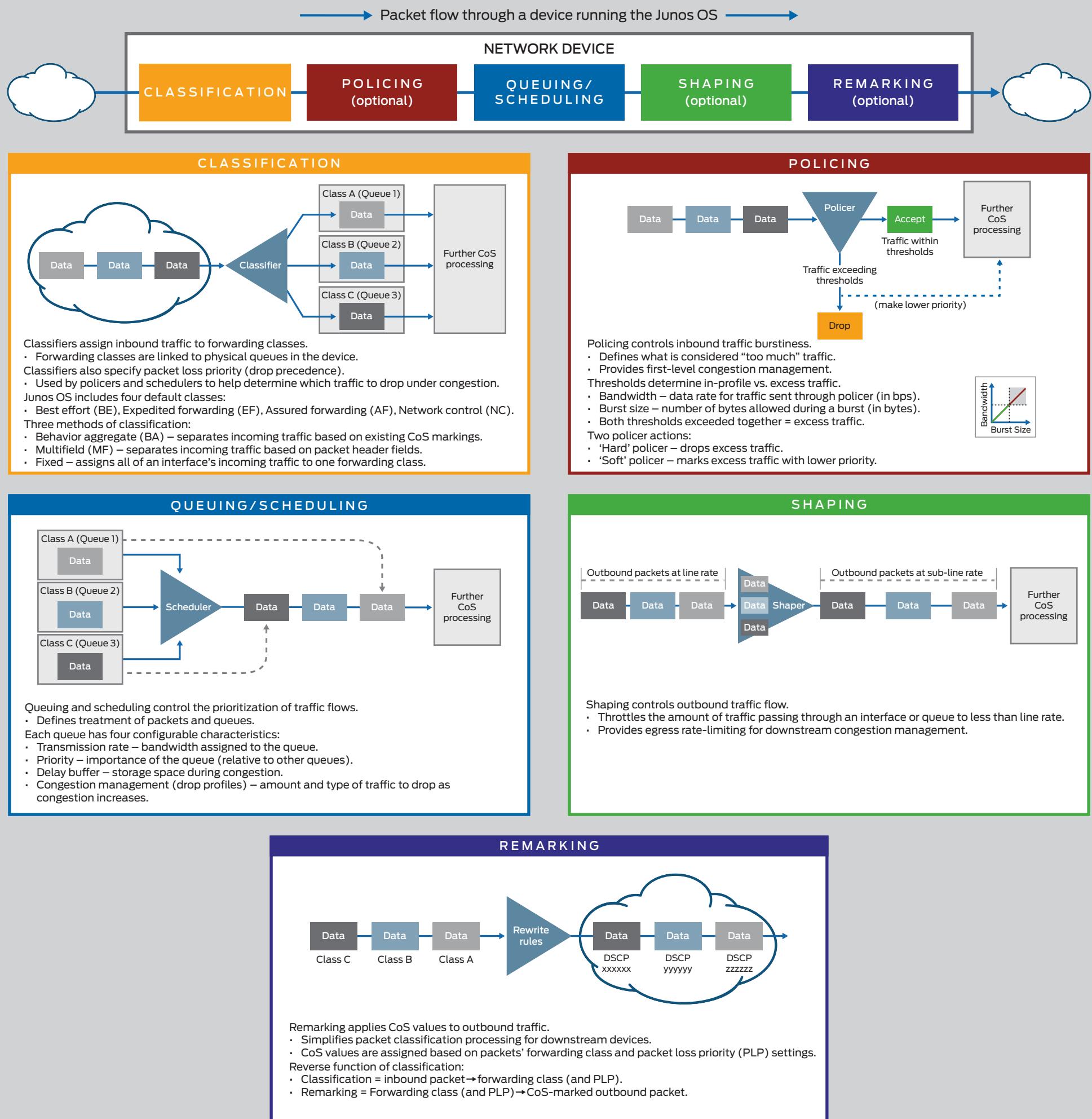
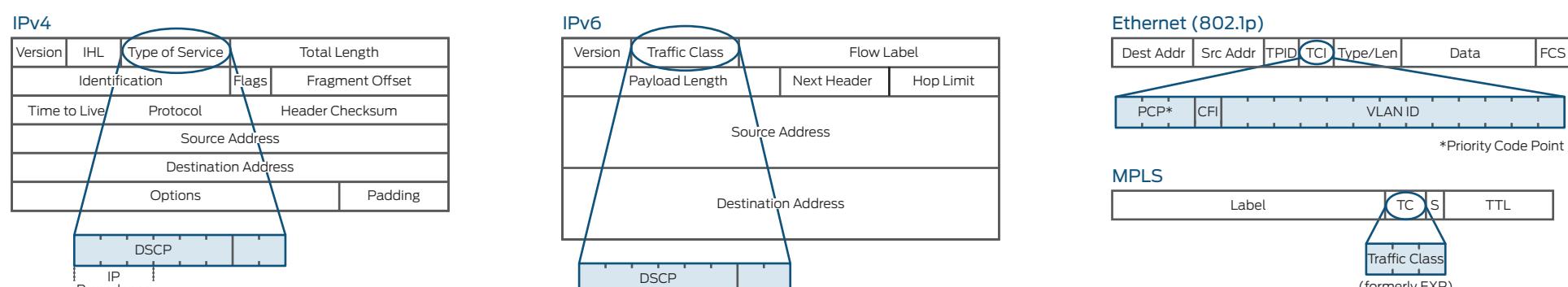


# Obtain Quality of Service with Junos® CoS

Junos CoS enables prioritization of traffic flows over a common path. It offers a means to recognize and control different types of traffic, provides the ability for applications to be considered more or less important, and provides mechanisms to manage congestion. Junos CoS is implemented “hop by hop”, at each device in the data path.



## CoS Fields in Packet Headers



For additional technical resources, please visit: [www.juniper.net/documentation/](http://www.juniper.net/documentation/)



Class of Service Feature  
Guide for Routing Devices



Learning Bytes Series:  
Class of Service Basics



Day One: Deploying  
Basic QoS



Day One: Junos QoS  
for IOS Engineers



Training Course: Junos  
Class of Service (JCOS)



Learn About:  
Quality of Service (QoS)

## DAY ONE POSTER

### Junos CoS

Juniper Networks  
Information and  
Learning Experience (iLX)  
[www.juniper.net/posters](http://www.juniper.net/posters)

# Configuring Junos® CoS

This example contains the main CoS configuration elements for a device running the Junos OS: forwarding classes, classifiers, policers, schedulers, shapers, and rewrite rules. Note that this example is for illustrative purposes only; while functional, there are many more configuration options and advanced settings not displayed here.

CLASSIFICATION

SHAPING

REMARKING

QUEUING/SCHEDULING

```

class-of-service {
    classifiers {
        dscp DSCP-BA-CLASSIFIER { ————— A
            forwarding-class Voice {
                loss-priority low code-points ef;
            }
            forwarding-class Data-1 {
                loss-priority low code-points af1;
                loss-priority medium-high code-points cs4;
                loss-priority high code-points cs3;
            }
            forwarding-class Data-2 {
                loss-priority low code-points af2;
                loss-priority high code-points cs2;
            }
            forwarding-class Data-3-BE {
                loss-priority low code-points be;
                loss-priority high code-points cs1;
            }
            forwarding-class Network-Control {
                loss-priority low code-points [ cs6 cs7 ];
            }
        }
    }
    drop-profiles {
        WRED-DROP-PROFILE-MODERATE { ————— B
            fill-level 75 drop-probability 20;
            fill-level 85 drop-probability 35;
            fill-level 95 drop-probability 50;
            fill-level 100 drop-probability 100;
        }
        WRED-DROP-PROFILE-AGGRESSIVE { ————— B
            fill-level 50 drop-probability 10;
            fill-level 70 drop-probability 40;
            fill-level 85 drop-probability 75;
            fill-level 100 drop-probability 100;
        }
    }
    forwarding-classes {
        queue 0 Data-3-BE;
        queue 2 Voice;
        queue 3 Network-Control;
        queue 4 Data-2;
        queue 6 Data-1;
    }
    interfaces {
        ge-* {
            scheduler-map SCHEDULER-MAP-A; ————— F
            unit * {
                classifiers {
                    dscp DSCP-BA-CLASSIFIER; ————— A
                }
                rewrite-rules {
                    dscp DSCP-REMARKING; ————— G
                }
            }
        }
        ge-0/0/1 {
            scheduler-map SCHEDULER-MAP-A; ————— F
            shaping-rate 800m;
        }
    }
    rewrite-rules {
        dscp DSCP-REMARKING { ————— G
            forwarding-class Voice {
                loss-priority low code-point ef;
            }
            forwarding-class Data-1 {
                loss-priority low code-point af1;
                loss-priority medium-high code-point cs4;
                loss-priority high code-point cs3;
            }
            forwarding-class Data-2 {
                loss-priority low code-point af2;
                loss-priority high code-point cs2;
            }
            forwarding-class Data-3-BE {
                loss-priority low code-point be;
                loss-priority high code-point cs1;
            }
        }
    }
    scheduler-maps {
        SCHEDULER-MAP-A { ————— F
            forwarding-class Voice scheduler Voice-Scheduler;
            forwarding-class Data-1 scheduler Data-1-Scheduler;
            forwarding-class Data-2 scheduler Data-2-Scheduler;
            forwarding-class Data-3-BE scheduler Data-3-Scheduler;
            forwarding-class Network-Control scheduler Nwk-Ctl-Scheduler;
        }
    }
    schedulers {
        Voice-Scheduler {
            transmit-rate percent 10;
            buffer-size percent 5;
            priority high;
        }
        Data-1-Scheduler {
            transmit-rate percent 10;
            buffer-size percent 10;
            priority medium-high;
        }
        Data-2-Scheduler {
            transmit-rate percent 40;
            buffer-size percent 25;
            priority medium-low;
            drop-profile-map loss-priority low protocol any drop-profile WRED-DROP-PROFILE-MODERATE; ————— D
        }
        Data-3-Scheduler {
            transmit-rate {
                remainder;
            }
            buffer-size {
                remainder;
            }
            priority low;
            drop-profile-map loss-priority low protocol any drop-profile WRED-DROP-PROFILE-AGGRESSIVE; ————— D
        }
        Nwk-Ctl-Scheduler {
            transmit-rate percent 5;
            buffer-size percent 5;
            priority high;
        }
    }
}

```

## Class of Service Stanza

Code Point Aliases (included by default on Junos devices) can be used in place of numeric values

user@device> show class-of-service code-point-aliases dscp

Code point type: dscp	Alias	Bit pattern
	af11	001010
	af12	001100
	af13	001110
	af21	010010
	af22	010100
	af23	010110
	af31	011010
	af32	011100
	af33	011110
	af41	100010
	af42	100100
	af43	100110
	be	000000
	cs1	001000
	cs2	010000
	cs3	011000
	cs4	100000
	cs5	101000
	cs6	110000
	cs7	111000
	ef	101110
	nc1	110000
	nc2	111000

CoS uses a modularized configuration model. Some elements are created and then referenced elsewhere.

● Create      ○ Reference

- A Apply the BA classifier to the desired inbound interface(s)
- B Apply the policers within the firewall filters (MF classifiers)
- C Apply the firewall filter (MF classifier) to the desired inbound interface(s)
- D Apply the drop profiles within the schedulers
- E Use a scheduler-map to apply schedulers to traffic classes
- F Apply the scheduler-map to the desired outbound interface(s)
- G Apply the rewrite rule to the desired outbound interface(s)

E

————— D

## Firewall Stanza

```

firewall {
    family inet {
        filter MF-CLASSIFIER { ————— C
            term VOICE {
                from {
                    protocol udp;
                    port 16384-32767;
                }
                then {
                    loss-priority low;
                    forwarding-class Voice;
                    accept;
                }
            }
            term INTERACTIVE-VIDEO {
                from {
                    protocol [ tcp udp ];
                    port 6060-6061;
                }
                then {
                    loss-priority low;
                    forwarding-class Data-1;
                    accept;
                }
            }
            term CALL-SIGNALING {
                from {
                    protocol tcp;
                    port 1720;
                }
                then {
                    loss-priority high;
                    forwarding-class Data-1;
                    accept;
                }
            }
            term NETWORK-MANAGEMENT-TELNET-SSH {
                from {
                    protocol tcp;
                    port [ telnet ssh ];
                }
                then {
                    loss-priority high;
                    forwarding-class Network-Control;
                    accept;
                }
            }
            term ICMP-POLICE {
                from {
                    protocol [ icmp icmp6 ];
                }
                then {
                    policer POLICER-ICMP-5M-DISCARD; ————— B
                    forwarding-class Data-2;
                    accept;
                }
            }
            term BEST-EFFORT-INTRANET-TRAFFIC {
                from {
                    protocol tcp;
                    port 8080;
                }
                then {
                    policer POLICER-250M-FC_D3-BE; ————— B
                    forwarding-class Data-2;
                    accept;
                }
            }
            term BEST-EFFORT-TRAFFIC {
                from {
                    protocol tcp;
                    port [ ftp ftp-data http https ];
                }
                then {
                    loss-priority high;
                    forwarding-class Data-3-BE;
                    accept;
                }
            }
            term all-else {
                then accept;
            }
        }
    }
    policer POLICER-ICMP-5M-DISCARD { ————— B
        if-exceeding {
            bandwidth-limit 10m;
            burst-size-limit 625k;
        }
        then discard;
    }
    policer POLICER-250M-FC_D3-BE { ————— B
        if-exceeding {
            bandwidth-limit 250m;
            burst-size-limit 625k;
        }
        then {
            loss-priority low;
            forwarding-class Data-3-BE;
        }
    }
}

```

## Interfaces Stanza

```

interfaces {
    ...
    ge-0/0/2 {
        unit 0 {
            family inet {
                filter {
                    input MF-CLASSIFIER; ————— C
                    address 192.168.1.254/24;
                }
            }
        }
    ...
}

```