# Configuring an efficient QoS Map

This document assumes the reader has experience configuring quality of service (QoS) maps and working with traffic prioritization. Before reading this document, it is advisable to consult and understand the document Configuring QoS in AOS, which can be found on the ADTRAN Support Community. This document is intended to be a guide, with tips and instructions on how to design QoS maps that efficiently use an AOS unit's processor, and do not create needless or redundant processing of traffic flows. Furthermore, this guide is intended for use with AOS units only, and is not a general guide for configuring QoS on other vendor platforms.

## When to Implement QoS

QoS is needed if there is a "bottleneck" in the network traffic path, which is caused by an interface or link being periodically congested or over utilized. A bottleneck is a link or interface in a network that has insufficient bandwidth to support the amount of network traffic allocated to traverse it at any given point of time. For example, a network consisting primarily of Ethernet ports will typically perform at a max of 100Mbps. If this network has a T1 link (1.544Mbps) as the wide area network (WAN) Internet connection, the T1 link would be a bottleneck for the rest of the circuit, because it does not support the same 100Mbps speed of the Ethernet traffic. WAN links are typically bottlenecks, but other interfaces or links can adhere to the same description and be bottlenecks as well.

A key point to remember when evaluating the need for QoS is how often a bottleneck link is congested. If the link is repeatedly congested, QoS may help, but this may be a sign that the link's bandwidth is not adequate. QoS will be a temporary fix in this case, when adding a higher bandwidth link is a long-term solution. Contrarily, if the link is never congested, QoS can add unnecessary processing to each packet, which can negatively affect an AOS unit's performance, based on the unit's processing ability and overall utilization.

## How to Design a QoS Map

The first priority when designing a QoS map is to identify the important types of traffic on your network and rank them in order of precedence. For example, there may be only a single type of traffic that is important, such as voice; or there could be multiple types, such as voice, Remote Desktop Protocol (RDP), and basic Internet traffic. Creating an ordered list, from most to least importance will help with the QoS design. Here is a list example:

SIP Voice Traffic and SIP signaling
RDP and FTP
SSH
Everything else

Notice there can be more than one type of traffic in each priority "queue," as they may be equally important. The less queues there are, the more efficient the QoS map can be, so this is good practice.

## How to Match Traffic Inside a QoS Map Efficiently

A QoS map needs to match traffic before it can make prioritization decisions. There are many different AOS match statements and the type and order of them is important to traffic flow efficiency. Below are rules to remember when configuring a QoS map.

Access-control lists (ACL) require considerably more processing than other match statements.

An ACL referenced by a QoS map requires the QoS map to call query the ACL for matches when traffic is being compared with that match statement. Once the ACL is queried, every rule inside the ACL has to be matched to a packet's parameters (for instance IP address, protocol, port, etc.) to look for a potential match. Below is an example configuration:

```
qos map VOICE 10
  match list VoiceACL
…continued…
!
ip access-list extended VoiceACL
  permit ip 192.168.1.0 0.0.0.255 any
  permit ip 192.168.2.0 0.0.0.255 any
  permit ip 192.168.3.0 0.0.0.255 any
```

If a packet with a source IP address of 192.168.3.1 attempts to traverse an interface with the QoS map "VOICE" applied, the packet will first be compared with the QoS map VOICE sequence 10. The QoS map sequence will query the "VoiceACL" ACL for packet parameter comparison. The packet will be compared with each line in the ACL. It will not match a permit statement until line three: "permit ip 192.168.3.0 0.0.0.255 any." Each individual packet from 192.168.3.1 will have to be compared with all three of "VoiceACL's" permit statements before it can be forwarded. An alternative configuration with greater efficiency would be if this traffic was simply marked with a DSCP value of 46/EF:

```
qos map VOICE 10
  match dscp 46
…continued…
```

Now when a packet with a source IP of 192.168.3.1 enters this QoS map, it will only need to be processed by one match rule, since it has been tagged with DSCP 46/EF prior to being processed by the AOS unit. This method prevents unnecessary processing by the AOS unit.

IP Precedence and DSCP can be redundant to a QoS map.

In the Type of Service (ToS) field of a packet, there are six bits relevant to QoS. The first three indicate an IP precedence value (0-7) and the six bits together indicate a DSCP value (0-63). These naturally convert in this manner:

*Table 1. IP Precedence Values and Their Corresponding DSCP Values*

| IP Precedence | DSCP |
|---|---|
| 0 | 0-7 |
| 1 | 8-15 |
| 2 | 16-23 |
| 3 | 24-31 |
| 4 | 32-39 |
| 5 | 40-47 |
| 6 | 48-55 |
| 7 | 56-63 |

*Note: IP Precedence to DSCP mapping is based on the binary number system which will not be covered in this document. Reference Configuring QoS in AOS for additional detailed information on this topic.*

For illustration purposes, the below example configuration is for a network that includes phones that can only be marked with an IP precedence value, while all other phones mark traffic with a DSCP value of 46. A DSCP value of 46 is the same as IP precedence 5, based on Table 1., so this QoS map only needs the statement **match precedence 5** because it will match both types of traffic marking.

```
qos map VOICE 10
  match precedence 5
…continued…
```

In a QoS map sequence with multiple match statements, the match statement that matches the majority of the traffic should be first in the configuration.

For example, say you want to match DSCP 46, for RTP voice traffic, and CS3, for voice signaling, in the first QoS map sequence because they are equally important. QoS maps, by default, treat each match statement within a sequence as a logical OR operation. In other words, in the below configuration example, if the traffic is marked as DSCP CS3, it will match the first statement and the QoS map is exited and the traffic is forwarded. If the traffic is not marked with CS3, it will then be checked to see if it is marked with DSCP 46 before it is forwarded.

```
qos map VOICE 10
  match dscp cs3
  match dscp 46
…continued…
```

In this situation, let's assume there will be more RTP traffic (DSCP 46) than signaling traffic (DSCP CS3). With the above QoS map, all of the RTP traffic will have to wait on the unit to process the **match dscp cs3** statement before progressing to the **match dscp 46** statement, which is what it will used to gain priority and be forwarded. In this situation, it would be more efficient to configure the QoS map as:

```
qos map VOICE 10
  match dscp 46
  match dscp cs3
…continued…
```

VOIP RTP packets (DSCP 46) will match the first rule and immediately be forwarded, where as the signaling packets (DSCP CS3), which make up a smaller percentage of the bandwidth, will undergo longer processing.

Take time to understand how traffic is marked throughout the network and that the markings are uniform at all sites.

As stated previously, the more match statements there are in a QoS map, the more processing power required to evaluate them. If the traffic that needs to be matched could possibly arrive at the AOS unit as DSCP 46 in some cases, but CS1 in another, DSCP 20 at a few sites, and IP Precedence 3 or 4 at others, this can make QoS maps inefficiently long and processor intensive. It is more efficient to take the time to mark all similar traffic flows with the same value, which will ultimately result in less processing time through QoS maps.

Avoid using the **match ip rtp <port range>** command.

When trying to match voice packets, it can be tempting to match network traffic based on the RTP port range. However, this not only matches the RTP ports, but also matches UDP packets with those port numbers. If there is a lot of UDP traffic on the network that could also use those ports, this is not efficient. It is recommend to tag the packets with an IP precedence or DSCP value and match based on that value instead of using the **match ip rtp <port range>** command.

## General QoS Configuration Guidelines

Avoid using the router to mark traffic unless necessary.

QoS maps that are used to mark a packet with a set command add additional processing to each packet, because they must match a certain type of traffic (DSCP, IP precedence, ACL, etc.) and then apply a new value to that packet. Traffic that doesn't need to be marked will still have to be checked by all of the match rules in the QoS map. For example, if you want to mark packets that make up 5% of the traffic that traverses the link, every packet will need to be checked, but only 5% will actually be marked. This can be an inefficient process.

While it's acceptable to mark traffic with an AOS unit, typically, the entity that generates the traffic has a more efficient way to mark the traffic for prioritization, because it will be able to naturally differentiate which traffic should be marked. For example, configuring a phone to mark the traffic it generates with a

DSCP value is easier than having the AOS unit inspect every packet it receives and evaluating it. The downside to having the end equipment mark the traffic is you must verify all the QoS trust boundaries are properly in place throughout the network so that incorrect entities cannot mark their traffic with a higher priority to get access to bandwidth they are not allocated for.

Avoid using "match-all" QoS maps.

Match-all QoS maps use logical AND decisions between all of the match statements in a QoS map sequence. When traffic triggers a match-all QoS map sequence, the traffic must match every rule in that sequence before it can be allocated to the actual queue bandwidth. For example:

```
qos map VOICE 10 match-all
  match dscp 46
  match vlan 10
  match list VoiceACL
…continued…
!
ip access-extended VoiceACL
  permit udp any any
```

For this example, the QoS map "VOICE" is designed to only perform action on traffic if the traffic is marked with DSCP 46, tagged as VLAN 10, and is UDP traffic. This QoS map could be used in an application where it is necessary to differentiate general voice traffic from voice traffic from VLAN 10. UDP traffic is referenced so that TCP traffic incorrectly marked as DSCP 46 in VLAN 10 could not be accidentally forwarded as part of this queue. Every packet that exits an interface with QoS map "VOICE" applied on it must be compared to every statement in it. This creates a lot of extra processing.

In this situation, a better QoS design would be to change the phones in VLAN 10 to mark their traffic with a different DSCP value than the other phones on the network. Furthermore, if the phones are the only devices in VLAN 10, the map can be created with one statement, **match vlan 10** because it would be the only type of traffic that can be tagged with VLAN 10.

There is a default queue implicitly at the end of every QoS map.

By default a QoS map has an implicit "match all" statement at the bottom and applies no bandwidth commands. This means whatever bandwidth is not allocated for the prioritized traffic up to the max-reserved-bandwidth (covered in the next section) can be used by the default queue. In this case, there is no need to create a "match-all" sequence at the end of a QoS map for all of the traffic that gets first in, first out (FIFO) forwarding, because it is there by default.

Increment the sequence number of a QoS map by 10 for each sequence.

Incrementing QoS map sequences by 10 is a scalability guideline that helps if a change ever needs to be made to a QoS map in the future. If the sequences are not separated, a sequence cannot be inserted between existing sequences, requiring the map to be completely re-configured.

## Assigning Queues Bandwidth

By default, 75% of an interface's bandwidth can be reserved for traffic. This is because bandwidth needs to be reserved for system-critical traffic, such as routing updates, status messages, and alarm information. The command that governs this is an interface level command, **max-reserved-bandwidth <percent>.** This can be adjusted up to 100%, but it is not recommended to apply this past 95%. Links that rely on keep-alive or negotiation traffic, such as PPP and Frame Relay, can go down if there is not enough bandwidth left over for link management traffic. Therefore, the **max-reserve-bandwidth 100** command should not be configured on an interface because it could cause link negotiation problems.

When creating a QoS map, there will be 75% (or what the **max-reserve-bandwidth <percent>** command is configured for) of the bandwidth reserved for priority traffic. You should follow the criteria below when deciding the sequential, top-down order or traffic importance. Each "queue" will be a sequence in the QoS map and can encompass more than just one type of traffic:

Most important queue first
Largest amount of traffic left
Largest amount of traffic left
Etc.

In the most important queue, use the prioritystatement for time-sensitive traffic (UDP, VoIP, Video), or the Bandwidth statement for normal traffic flows with their own retransmission mechanisms (TCP, Data). In a QoS map, there is one priority queue that will always forward traffic until it's empty before other queues are able to forward traffic. This is where the most time-senstitive traffic on the network should be referenced, especially if it is using UDP or a similar protocol that does not support retransmission Unless a very specific amount of bandwidth (to Kbps) is known for this queue, it is easiest to use the priority percent <%> command to give this queue a percent of the bandwidth.

If the traffic in the top most sequence is not time sensitive, use a normal **bandwidth percent <%>** command so that is gets queued up properly with the other traffic queues. If you use a priority statement, the map will assume the traffic is time-sensitive. If there is not enough available bandwidth to forward the traffic, it will immediately be dropped whereas using the bandwidth command will cause it to be delayed. This is because things like voice and video would have more problems with delayed traffic than if packets were just dropped periodically.

In the rest of the sequences, the **bandwidth percent** command can be used until the total reserved bandwidth adds up to the max-reserve bandwidth minus the percentage you want to leave for the traffic in the default queue:

**Seq1 bandwidth + Seq2 Bandwidth +…+ Default queue Bandwidth = Max-Reserve-bandwidth**

The reason to have the largest queues towards the top of the map (not including the priority queue) is so fewer packets will traverse the entire map. For example, if the largest queue is the last of five map sequences, the majority of traffic will have to be processed by the entire QoS map, which is inefficient.

Furthermore, if the default queue is much bigger than the rest of the priority queues combined, it may be better to look into another measure of limiting traffic coming from different sources over making the default traffic traverse a large QoS map as this would mean the majority of your traffic has to be thoroughly processed by the unit.

Another guideline is to avoid using the command **priority percent unlimited**. If you want the traffic to have priority over everything else, it may still be worthwhile to give it a percent of bandwidth as you cannot see QoS map drop statistics for other queues when using the **priority percent unlimited** command.

The **Strict-priority** command is also a command that should be generally avoided. This command actually removes a piece of the bandwidth for the sequence, so that it cannot be used by anything else, and that type of traffic can never use more than that piece of bandwidth. By default, QoS rules will only come into play when the link gets congested, so unless you specifically want to permanently reserve a piece of the bandwidth away from all other traffic, you should not use the **strict-priority** command. In this scenario, if a burst of traffic matching the map came through that was larger than the value you had set in the sequence, the extra traffic would be dropped whether there was bandwidth available for it or not.

## Troubleshooting QoS Maps

Typically, there are only a few **show** commands to use when troubleshooting a QoS map. The first is **show qos map interface <type> <slot/port>** shown below.

```
 # show qos map interface vlan 1

 vlan 1
  qos-policy out: NAME

   map entry 10
     match ip list any
     priority bandwidth: 50 (% of total)
     burst budget 1250000/1250000 bytes (current/max)
     packets matched: 0, bytes matched: 0
     packets dropped: 0, bytes dropped: 0
     5 minute offered rate 0 bits/sec, drop rate 0 bits/sec

   map entry default
     packets matched: 62, bytes matched: 3941
     packets dropped: 0, bytes dropped: 0
     5 minute offered rate 0 bits/sec, drop rate 0 bits/sec
```

The important sections of the output are the “packets matched” and “packets dropped” sections. The data above that is an indication of your configuration, as long as you have verified your QoS design, that portion of the output will not be relevant to your troubleshooting.

This command will present a section of output for each sequence of the QoS map that is assigned to the interface specified. The “packets matched” field can be used to indicate if the QoS map is matching any packets. If there are zero matches, then either the configuration is incorrect, or the traffic may not have the characteristics the QoS map is configured to match (i.e. DSCP value, IP address, VLAN tag, etc.). In this case, it is recommended to check the source equipment’s configuration, or to use a packet capture to verify packet parameters.

Furthermore, this command’s output can be valuable if you are experiencing quality issues and probable packet loss with a certain type of traffic. In each sequence of the QoS map, you will see:

packets dropped: 0, bytes dropped: 0
    5 minute offered rate 0 bits/sec, drop rate 0 bits/sec

This output indicates if the traffic matched in this QoS map sequence is being dropped by the AOS device. If traffic from a certain sequence is being dropped, then the available bandwidth in that QoS map’s sequence may need larger bandwidth to account for extra traffic. Similarly, if a sequence displays a very low number of matches along with zero drops consistently, it may be possible to decrease this sequence’s bandwidth to give more available bandwidth to other important applications.

The last section of the output is the “default” queue. You should see the majority of the dropped traffic in this sequence, as this traffic is not prioritized:

map entry default
    packets matched: 6245, bytes matched: 39412342
    packets dropped: 1023, bytes dropped: 8048939
    5 minute offered rate 124364 bits/sec, drop rate 25634 bits/sec

Examining the above “drop rate,” this QoS map is (on average) dropping 25634 bits of traffic per second. This, however, may not be an indication of a problem. If users or applications are not experiencing a noticeable issue, then this traffic is probably being retransmitted and eventually completing (for example, research TCP retransmissions). If an application is experiencing problems, the best remedy would be to decrease the bandwidth in other sequences so there is more bandwidth for the default queue. Another solution would be to make this application prioritized instead of leaving it in the default sequence.

Large amounts of drop rates in the default sequence, paired with the other queues showing low numbers of matched packets, does not necessarily indicate a need to change QoS match criteria. QoS only comes into effect when the link is congested. A large drop rate in the default sequence, and low numbers of matched packets can indicate that the link does not have enough bandwidth for the amount of traffic that is trying to traverse it. In this case, evaluate what types of traffic are traversing the network and decide if the allocation of bandwidth needs to be addresses.

## Important QoS Configuration and Troubleshooting Tips

This section is a collection of tips to use when diagnosing problems with prioritization and packet loss of priority traffic.

QoS maps only take affect when the link is saturated.

QoS rules are a guideline the router uses to prioritize important traffic if inadequate bandwidth on a link requires traffic to be delayed or dropped. Unless specifically using the **strict-priority** command, reserving bandwidth for a certain type of traffic will not make that portion of the reserved bandwidth unavailable to other traffic types. Similarly, the portion of reserved bandwidth is not a limit on the priority traffic type if there is non-reserved bandwidth available.

On Ethernet and VLAN interfaces, traffic-shaping is required for QoS to take effect.

Ethernet interfaces and VLANs natively run at wire speed on an AOS unit. For example: A metro Ethernet connection may negotiate Ethernet at 100Mbps, but the carrier may limit the available bandwidth to an upload rate of 20Mbps. In this case, Ethernet interfaces will assume that they have 100Mbps of bandwidth available to calculate QoS queue depth. If the link is not shaped down, QoS rules will not take affect because the link will never be fully congested (at 100Mbps) from the AOS device's perspective. In this case, drops will not increment on the router, but the service provider will be dropping traffic (without any prioritization) upstream.

To enable traffic shaping, do so at the interface level with the command **traffic-shape rate <rate in bps>**. This is not required on non-broadcast interfaces as they will either negotiate bandwidth or interpret it from the layer 1 interface cross-connection (T1, E1, etc.). Here is an example configuration of traffic-shaping an Ethernet interface to 20Mbps:

```
interface eth 0/2
  traffic-shape rate 20000000
***continued***
```

Incoming traffic cannot be shaped or prioritized.

Traffic that is received on an interface cannot be shaped or prioritized (though it can be matched and marked) as it has to be fully processed by the unit before decisions of that nature can be made. The only thing that controls prioritization of received traffic is the sending unit. When applying traffic shaping and QoS map prioritization to an interface only sent, or egress, traffic is affected. Traffic that requires a two-way conversation stream, for instance voice or video conferencing, must be prioritized at both ends of the link.

For example: The company Service Inc. has phones that must be able to make and receive calls to the outside world through their provider, Voice Provider LLC. In this case, voice needs to be prioritized so that data bandwidth spikes will not interfere with voice traffic flows. In this case, Service Inc. sets up prioritization for voice traffic on their outgoing AOS unit's interface.

However, Voice Provider LLC. is not aware that Service Inc. needs voice prioritization on the link and runs with a default configuration. Service Inc.'s customers can always hear them clearly throughout the call, but the employees hear poor voice quality during congested periods of the day. The outside world

hears Service Inc. fine because outbound traffic is prioritized in the AOS unit. However, the voice traffic Service Inc. receives is not prioritized by the provider and therefore it is being transmitted with drops or delays.

Furthermore, outbound traffic has no effect on inbound traffic. If the link is saturated from inbound traffic, it doesn't matter how the AOS unit prioritizes outbound traffic, because the link is already full. The link will not be able to accept additional traffic and poor quality may result. Therefore, it is recommended to have the upstream device provide QoS as well.

Default Internet routers will not prioritize traffic.

If you have priority traffic that routes out an Internet circuit, it is important to note that QoS on your outbound WAN interface (the bottleneck), will not ensure that the traffic's recipient will receive the packets in a timely fashion. QoS only controls decisions on the link it is applied to. If there are 20 more hops with possible bottlenecks between the sender and the destination, all of these links would need to prioritize the traffic type in times of congestion to ensure proper delivery. Typically, Internet routers will not do this. To get prioritization across the Internet, the service provider must have a dedicated line for you (for instance, a T1, T3, or a point-to-point circuit between offices). QoS will help congestion on the WAN router it is applied to, but it may not resolve congestion issues when using a public Internet circuit.

Typically, in a private network, QoS is not needed at each hop.

A common misconception is that QoS is required on every device in a network. This network design can add excessive processing to traffic flows unintentionally. As previously stated, QoS maps only need to be implemented on "bottleneck" links.

For example, there are three distribution switches that have 1Gbps links to every section of a building. However, when adding a new office, the only equipment available has a maximum transmission rate of 100Mbps. It can be calculated that this office could transmit and receive over 200Mbps of bandwidth at any point in time. In this case, it would be wise to implement QoS on both ends of this link (outbound only) to make sure the office's priority traffic is always routed through. Adding QoS on all of the other 1Gbps links would create unneeded processing as the 200Mbps is less than the other link's bandwidth of 1Gbps.

VPN traffic can only be prioritized by DSCP or IP Precedence.

When prioritizing traffic that is leaving an encrypted interface (over a VPN), the only way to match traffic is by DSCP value or IP Precedence because the rest of the packet is encrypted. If the originating traffic cannot mark itself with a DSCP value or IP Precedence, then you can create an inbound QoS map on the interface that the traffic enters the AOS unit and mark the traffic with a DSCP/IP Precedence value that is not used anywhere else in the network. Then on the outbound interface's QoS map, match the value that was set on the inbound QoS map.

In the example below, QoS map "QOS-VPN-TRAFFIC-INBOUND" is created to match an ACL "VPN-SUBNET" and set that traffic with an arbitrary DSCP value of 56. Then the QoS map "QOS-VPN-TRAFFIC-INBOUND" is applied on the inbound interface. Here is the example configuration:

ip access-extended VPN-SUBNET

```
  permit ip 10.1.1.0 0.0.0.255 10.1.2.0 0.0.0.255
!
qos map QOS-VPN-TRAFFIC-INBOUND 10
  match ACL VPN-SUBNET
  set dscp 56
…continued…
!
interface eth 0/1
  qos-policy in QOS-VPN-TRAFFIC-INBOUND
```

Now that the traffic that will be encrypted and sent over the VPN was marked with DSCP 56 on the inbound interface, DSCP 56 can be matched and given priority on the outbound interface, as shown in the example below:

```
qos map QOS-VPN-TRAFFIC-OUTBOUND 10
    match dscp 56
…continued…
!
interface ppp 1
  qos-policy in QOS-VPN-TRAFFIC-OUTBOUND
```

## Constructing an Efficient QoS Map Example

The following is an example of how to create an efficient QoS map in AOS using the guidelines from this document.

Company Inc. has a dedicated point-to-point link between their two offices, which is made up of eight T1s (or 12Mbps bandwidth). Each individual office's LAN is made up of 100Mbps links. Traffic that is sent between the two offices can reach an excess of 20Mbps at congested periods. Company Inc.'s network administrator decides that both sides of this link will need QoS, so traffic can be properly prioritized between sites. The map will consist of four queues/sequences:

SIP Voice Traffic and SIP Signaling
RDP and FTP
SSH
Everything else

The SIP voice traffic (RTP) and SIP signaling are the most important traffic in the network, and since they are delay sensitive he will create a LLQ for this traffic with the **priority** statement. Furthermore, he has deduced that it will account for up to 50% of the traffic on the link in times of congestion. RTP packets will be marked with DSCP 46 by the phone system, and SIP signaling packets will be marked with DSCP CS3. However, some of the older voice equipment can only tag SIP signaling packets as DSCP 41 and cannot be changed. This will make up a small portion of the traffic. RTP packets marked with DSCP 46 will make up the majority of the traffic in this sequence.

The first sequence (qos map Prioritize_Traffic 10) will have three match statements and will use the default logical OR operation for decision criteria. Therefore, traffic that matches any of the three types specified in the QoS map will immediately be applied to the priority statement without processing anymore of the QoS map. Since RTP packets (DSCP 46) make up the most traffic, it will be the first option, CS3 signaling will be the second, and the old voice signaling traffic will be the third (DSCP 41), since it encompasses the least amount of traffic. The first QoS map sequence would look like the example below:

```
qos map Prioritize_Traffic 10
 match dscp 46
 match dscp cs3
 match dscp 41
 priority percent 50
```

The second sequence (qos map Prioritize_Traffic 20) is for RDP and FTP applications. This traffic is expected to encompass up to 20% of the link bandwidth during times of congestions. RDP uses TCP port 3389 by default, but the network administrator has set all the computers to mark RDP traffic as DSCP 20. FTP could not be marked with a DSCP value, but is always originating from VLAN 5 in each office's network.

In this sequence, RDP traffic (DSCP 20) will be the first match statement since it is used more often, anFTP (VLAN 5) will be matched second. This traffic will use a **bandwidth** command since it is delay insensitive; and it will be allowed to use up to 20% of the bandwidth when the link is congested. This QoS map sequence would look like the example below:

```
qos map Prioritize_Traffic 20
 match dscp 20
 match vlan 5
 bandwidth percent 20
```

The third sequence in the QoS map is for an SSH application between two servers, one in each office. This connection is always on, and by default, is marked with IP precedence 3. This connection uses about 5% of the bandwidth.

```
qos map Prioritize_Traffic 30
 match ip precedence  3
 bandwidth percent 5
```

The final QoS map will look like this:

```
qos map Prioritize_Traffic 10
 match dscp 46
 match dscp cs3
 match dscp 41
 priority percent 50
!
qos map Prioritize_Traffic 20
 match dscp 20
```

```
  match vlan 5
  bandwidth percent 20
!
qos map Prioritize_Traffic 30
  match ip precedence 3
  bandwidth percent 5
```

## QoS FFE feature

As of R10.6.0, QoS has been added as a feature that benefits from FFE, which can be understood more by reviewing this document: RapidRoute/FFE in AOS. This allows the unit to store session information on QoS within FFE and then match to session information for packets in a flow, instead of reclassifying each individual packet. It is recommended that with very large QoS maps, upgrade your unit to R10.6.0 or higher.