ServiceQ

White Paper by Aperto Networks

Subir Varma

Version 1.1
June 7, 2002

Table of Contents

1 Introduction 3
2 ServiceQ Features 3
3 Classifiers 4
4 Traffic Shaping 5
4.1 Reasons for Using Traffic Shaping 6
4.2 Token Bucket Algorithm 6
5 Classes of Service. 7
5.1 Weighted Fair Queuing 8
5.2 System Capacity and Admission Control. 9
5.3 Over-subscription 9
5.4 Voice Support 10
6 Configuring ServiceQ Parameters 10
6.1 BSU Configuration Parameters 11
6.2 SU Configuration Parameters 12
6.2.1 Using a CIR Service Flow to specify a minimum guaranteed bit rate 12
6.2.2 Using Traffic Shaping to specify a maximum allowed bit rate 18
6.2.3 Advanced Parameters 19
6.2.4 Voice Support 19
7 Troubleshooting FAQ 22

1 Introduction

Aperto Networks PacketWave® (PW) system has been designed to provide sophisticated quality of service (QoS) capabilities to the end users and the service providers. All incoming traffic is classified into service flows and a scheduling mechanism operating at the MAC layer is responsible for assigning bandwidth to these flows according to their class of service and their QoS requirements. This document walks the reader through all of PacketWave's diverse offerings related to QoS and shows that while it may appear complex, PacketWave is also as simple as it should be.

2 ServiceQ Features

Quality of Service is all about parceling up bandwidth for various applications based on their requirements. This means that the traffic must be categorized and separated into different flows, which can then be prioritized accordingly. Aperto's ServiceQ technology performs this service. Classifiers divide up and queue the traffic onto different service flows (each of which can have different QoS requirements) as shown in Figure 2-1.

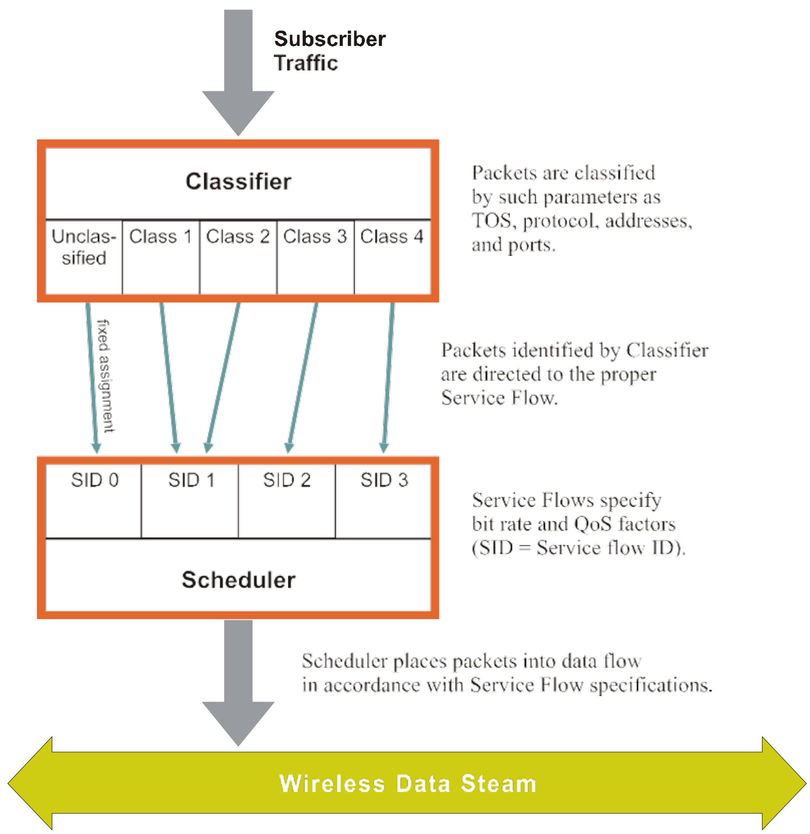


Figure 2-1: PacketWave traffic classification to service flows

A Subscriber Unit (SU) can have all of its traffic go over a single service flow or divide its traffic among several service flows. Each service flow is represented by a unique service flow ID (SID) and uses a separate queue. Per-flow queuing is an important feature of the PacketWave systems since it protects delay-sensitive service flows from extensive queuing delays and packet loss, which would be caused if a single shared queue had been used.

Service flows always come in pairs: one for upstream and one for downstream. Note that upstream service flows can have some differences in QoS configuration from the downstream, and the upstream and downstream classifiers can also be configured to match different traffic flows, if so desired. However, a single identical pair of upstream and downstream classifiers matching all traffic for the SU should suffice for most situations, with perhaps an additional pair of classifiers for voice.

For some more background on Aperto Networks' ServiceQ technology, you can refer to Chapter 2 of the *PacketWave 1000 Series Base Station Equipment Installation, Configuration, and Operation Manual*.

3 Classifiers

The purpose of packet classifiers is to direct incoming traffic to the appropriate service flows based on the classification rules defined by the users (as shown in Figure 2-1). The classifier rules are specified in the SU configuration file, which is downloaded from the TFTP server during SU initialization procedure. During initial registration, the SU sends the downstream classification rules to the BSU, which then maintains them in a table. The downstream classification rules are executed on the BSU and apply to all incoming traffic on the BSU Ethernet interface. The SU also maintains a separate table for the upstream rules. The upstream rules are executed on the SU and apply to all incoming traffic on the SU Ethernet interface.

Each classification rule may be based on layer 2, layer 3, or layer 4 elements or any combination thereof, depending on the requirements of the network traffic. The rule items can be defined from the list of the following elements:

a) Source MAC address
b) Destination MAC address
c) IP TOS (Type Of Service)
d) IEEE 802.1p user priority
e) IP Protocol (e.g. TCP, UDP, or ICMP)
f) Ethernet type / 802.2 SAP
g) Source IP address (and subnet mask)
h) Destination IP address (and subnet mask)
i) Protocol source port number (a range may be specified)
j) Protocol destination port number (a range may be specified)

Multiple items from the above list may be selected by a single classification rule; in that case an incoming packet will meet the rule only if a logical AND condition among the selected elements is satisfied. For every service flow that is created (except the default Best Effort flow), at least one classifier rule must be defined. Each classifier rule is assigned a unique priority value, which is necessary whenever multiple rules are defined for a single flow. In the event of multiple rules, they are executed in the order of their priority (the rule with the higher numbered priority gets the higher priority). If an incoming packet meets a certain rule, lower priority rules are ignored. Otherwise the next immediate lower priority rule is checked until all rules have been examined.

The left portion of Figure 3-1 shows a simple example of a classifier, which would be used to classify voice traffic to service flow SID 1 (presumably, one configured for voice, refer to section 5 and 6.2.4) and TCP traffic to service flow SID 2. All other traffic not matching any of the classification rules would be served by the default flow SID 0. In that case, we could possibly use the IP addresses of the voice gateways as the source and/or destination IP addresses of the classifier 0 rules, and the protocol type (TCP in our case) for the rules of classifier 1. If classifier 0 had a higher priority (priority 20), each arriving packet would be compared against classifier 0 first and if it matched the rules, it would be handled by the SID 1. If it didn't match the rules of classifier 0, it would be compared against the classifier 1 rules, and if it matched, it would be sent to SID 2. All traffic not matching either classifier 0 or 1 rules would be served by the default SID 0. The flowchart on the right portion of Figure 3-1 illustrates this example (note you can have more classifiers than this, even multiple classifiers with different priorities mapped to the same service flow).

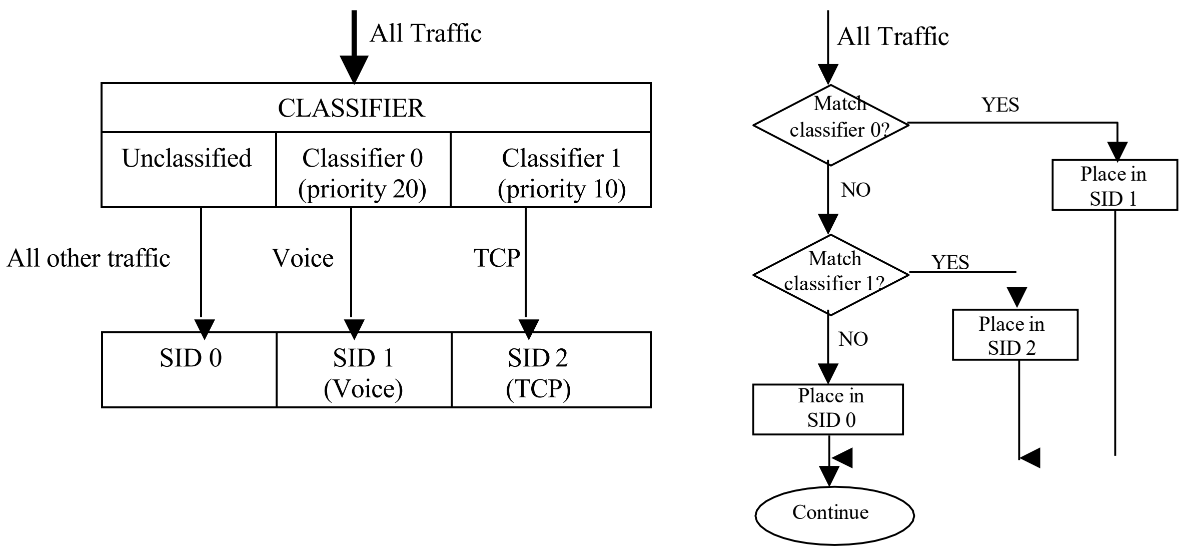


Figure 3-1: Classifier example

In section 6 we provide additional details on how to configure a service flow and create classifier rules.

4 Traffic Shaping

Both downstream and upstream service flows can be configured for peak-rate traffic shaping, which will instruct the PacketWave system to limit the bit rate available to that flow. If that flow tries to send traffic faster than the configured peak rate, the PacketWave system will delay packets to keep the flow within the configured rate. If too many packets are being delayed, then the oldest one (head-of-line) will be discarded (this has the advantage of causing TCP traffic to detect and back off sooner). In addition, the amount of burstiness allowed is also configurable via the token bucket size, which will be discussed in more detail in section 4.2.

4.1 Reasons for Using Traffic Shaping

PacketWave's WFQ-based scheduling already ensures that no single SU or flow will hog too much bandwidth to starve other flows or prevent them from meeting their minimum requirements. However, misbehaving service flows can still impact the performance of other flows, and traffic shaping can be used to prevent this.

Traffic shaping can also be used to offer different tiers of service. For example, a basic level of service can come with a service flow with traffic 64 kb/s traffic shaping, and a higher level of service can offer a flow with 1024 kb/s shaping, and a still higher level can offer a flow with no peak rate shaping, all of which can be priced accordingly.

Another benefit is to provide more consistency in level of service. Users who experience the service both during peak load and when there is very little load will see a very large discrepancy which can result in a more frustrating experience than for a user with traffic shaping who gets a more consistent level of service – even though on average their average bandwidth may be *lower* than the first user.

Note that currently available third-party equipment to perform traffic shaping can only do so in the downstream direction, whereas PacketWave already offers it in both the downstream and upstream direction.

4.2 Token Bucket Algorithm

PacketWave uses the token bucket algorithm to perform traffic shaping. The system puts tokens into a flow's bucket at a fixed rate (the peak rate), and the flow "spends" one token from the bucket per byte that it wants to send. However, the maximum number of tokens the flow can amass is limited by the token bucket size, meaning that once the bucket is full, the flow does not receive any more tokens. If a packet arrives when there are enough tokens for it, then the packet is queued for transmission. If there are not enough, then the packet waits until enough tokens accumulate. This is illustrated below in Figure 4-1 (for the sake of simplicity, the figure does not show realistic packet sizes nor a realistic number of tokens required per packet, but just illustrates the basic idea).

Essentially, this means that a flow that constantly sends traffic below or at the peak rate will be able to transmit all of its packets without delay, but a flow that tries to exceed the peak rate will have some packets delayed. In addition, the system allows burstiness, limited by the token bucket size. Basically, a flow which stays *below* the peak rate (e.g. it's idle) can "save up" tokens allowing it to temporarily exceed the peak rate later. The token bucket size is the limit of how many tokens it can "save up," thus limiting the size of the burst.

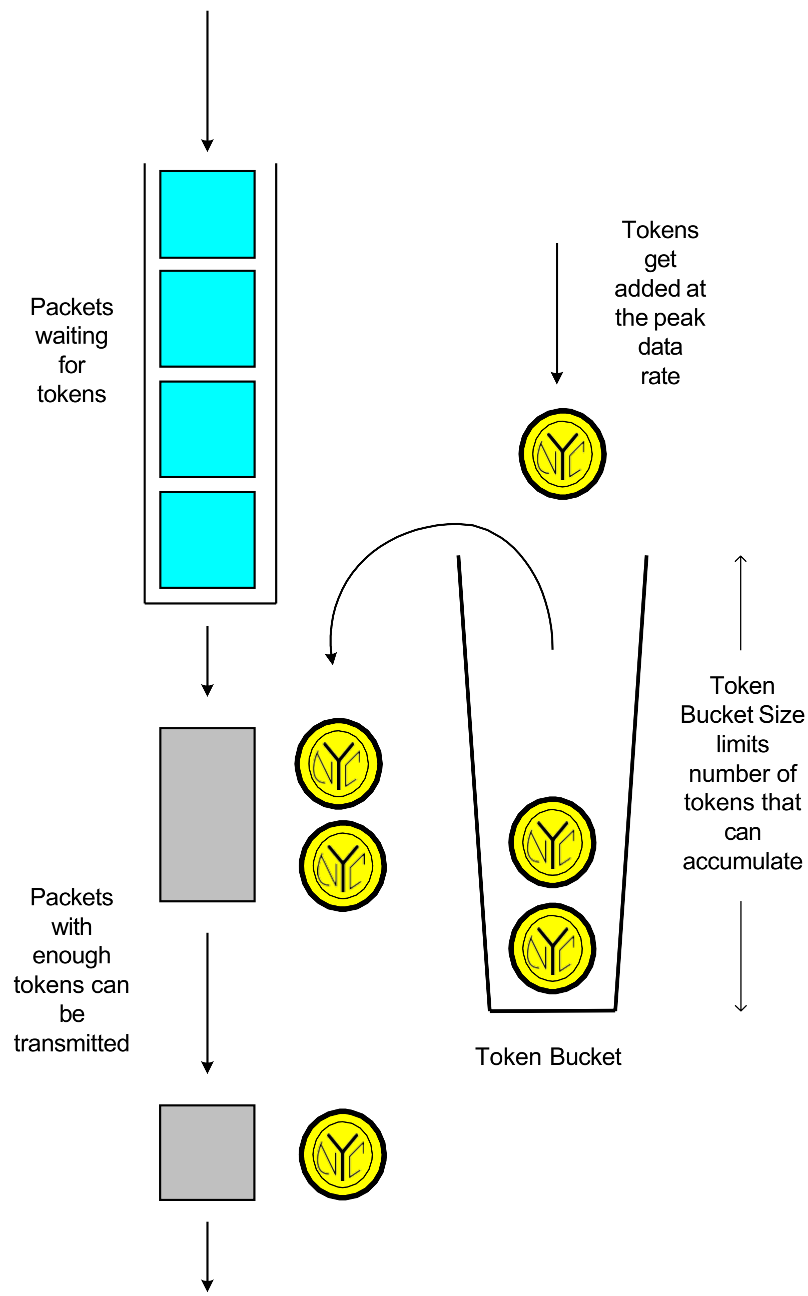


Figure 4-1: Token Bucket algorithm

5 Classes of Service

The PacketWave system offers service flows in the following three Classes of Service (CoS):

- **Best Effort (BE)** service flows are not given any performance guarantees, and are only allocated bandwidth after the requirements of the other service classes are satisfied (this is the class of service given to most current residential DSL and cable modem users). However, to prevent starvation, the *group* of all BE flows is guaranteed a configurable amount of bandwidth. Each SU has a default BE service flow that cannot be changed to any other CoS.

- **Committed Information Rate (CIR)** flows are given a guaranteed minimum amount of bandwidth, and can be offered as a higher-priced tier of service. Note that the configured bandwidth is a minimum that can be exceeded if extra bandwidth is available (in contrast to traffic shaping, which enforces a maximum).
- **Constant Bit Rate (CBR)** flows are allowed to send fixed-size packets at a recurring interval with as little latency as possible, and thus are suitable specifically for voice and other applications with similar real-time requirements.

One way to look at the different classes is like freeway traffic where you have express toll lanes (where you pay an extra fee to be able to use a faster lane during rush hour). These lanes are like CIR flows (except that the lanes don't actually guarantee a speed, like CIR does). The normal lanes are like BE flows. CBR is like… well, it's like staying off the freeway altogether and using your phone!

Note that traffic shaping is only configurable for CIR and BE flows (including the default BE service flow). CBR flows automatically use traffic shaping at their configured constant bit rate, and this cannot be changed.

5.1 Weighted Fair Queuing

To satisfy CIR requirements while preventing starvation of BE flows, PacketWave employs a version of Weighted Fair Queuing (WFQ). Basically WFQ seeks to approximate a GPS (Generalized Processor Sharing) system, which sends a bit at a time from each flow, and multiple bits for flows that are of higher priority (higher weighted). Bit-by-bit transmission isn't feasible, so most queuing algorithms attempt to approximate the outcome by transmitting packets in the same order they would finish transmitting in the GPS system. Exactly duplicating the order would require too much time to calculate, so algorithms like WFQ just try to approximate it. Since individual BE flows receive no guarantee, they do not receive a weighting, so the PacketWave system groups them together and weights the entire BE group with the bandwidth not allocated to the CBR and CIR classes.

Moreover, in order to use this weighting to provide an actual bandwidth guarantee, PacketWave employs admission control, discussed in more detail in the next section, to ensure the total bandwidth of admitted CIR flows does not exceed the total amount of bandwidth allocated to the CIR class. This means that weighting each admitted flow will give each flow their minimum guaranteed bandwidth if the entire CIR allocation is used up, and more bandwidth if there is extra bandwidth available.

In addition to providing bandwidth guarantees, WFQ enables very efficient utilization of the available bandwidth by allowing active flows to take advantage of the excess bandwidth at any given time. For example, if a CIR flow is not using up its minimum guaranteed rate (or is completely idle), then the unused bandwidth can be used by other flows. These can be either other CIR flows that can then exceed their minimum rate, or the group of BE flows. In fact, the excess bandwidth is dynamically allocated in proportion to the weights of each active CIR flow and the weight of the BE group (provided at least one BE flow is active). If, however, the CIR flow of the example bursts up to a higher rate, then it instantaneously gets back its reserved bandwidth.

5.2 System Capacity and Admission Control

The PacketWave system has a finite amount of wireless bandwidth available, so it obviously can fulfill the guarantees for only a finite amount of bandwidth to CBR and CIR service flows. In order for the system to track this, the service provider must specify what percentages of the total available bandwidth should be allocated to the CBR, CIR, and BE classes.

The amount of total upstream and downstream bandwidth available per sector depends on the channel width, TDD frame size, upstream/downstream allocation, and the number of REQ and ACK slots per TDD frame. For details on the available bit rates per channel width please refer to and also to *Aperto Networks White Paper on WMAC Parameters.* When configuring the BSU, this total bandwidth must be partitioned among the CBR, CIR, and BE classes.

CBR and CIR flows are then subject to admission control. For example, when a CIR flow tries to send traffic, the PacketWave system determines if there is enough available CIR bandwidth to accommodate this CIR flow. If so, the service flow is admitted and awarded a bandwidth reservation, and the total available CIR bandwidth is decreased by the amount of the reservation. When the service flow becomes idle for longer than a configurable timeout, its bandwidth reservation is returned to the available CIR bandwidth. However, if there is not enough available CIR bandwidth, then the service flow is denied admission and is not given a reservation. Instead, it temporarily gets BE service until enough CIR bandwidth is made available by other CIR flows releasing their reservations.

Likewise for CBR flows. BE flows are not subject to admission control, since they have no bandwidth guarantees. However, as mentioned before, the system needs to know what percentage of bandwidth to allocate to BE flows in order to use WFQ and prevent starvation.

5.3 Over-subscription

ServiceQ allows the system to permit configurations of CIR and CBR flows with more combined total configured bandwidth than is available in the system, i.e. over-subscription. For example, if the downstream bandwidth allocated to CIR is 8 Mb/s, and the service provider configured 24 different users with a downstream 1 Mb/s CIR service flow, then this is oversubscription by a ratio of 3:1.

If there are never more than 8 users simultaneously active at a given time, then everybody's committed rate can be satisfied. However, if a 9th CIR flow goes active while the other 8 are still busy, the PacketWave system will give that user BE service, and thus the already admitted CIR flows will continue to have their bandwidth guarantees met. You can liken this to when an airline overbooks: if too many passengers show up, then some have to be bumped to a later flight. In addition, like an airline, the service provider can realize more revenue by oversubscribing appropriately.

Admission control works seamlessly with the Adaptive Coding and Modulation scheme, such that the bandwidth guarantees continue to be met even when the modulation or coding for the link changes. For example, a 1 Mb/s CIR flow whose SU's link drops from 16QAM to QPSK will end up taking up 2 Mb/s worth of the available bandwidth, and the admission control algorithm will account for this accordingly. Since this can increase the bandwidth being used, providers should take this possibility into account when oversubscribing.

ServiceQ can continue to meet bandwidth guarantees even with a high degree of oversubscription (3:1 and higher) because link bandwidth is only reserved for those flows that are actively sending traffic. If a flow becomes idle for more than 100 ms (this is configurable — refer to section 6.2.3), then its reservation is taken away. User activity while doing web surfing for example, alternates between periods of activity interspersed by silence, which allows ServiceQ to quickly and efficiently swap users in and out of the reserved bandwidth class, while giving each of them their guaranteed rate when they are active. For example, if a user spends 5 seconds downloading a web page (which average about 20 Kbytes), and then spends the next 30 seconds reading it, then the system reserves bandwidth for him for only the 5 seconds that he is actually transferring the data. The rest of the time, his bandwidth can be used by other active users.

5.4 Voice Support

The PacketWave system provides support for Voice-over-IP (VoIP) applications via the CBR class of service. It is well known that voice quality in VoIP packet networks is very sensitive to packet latency and jitter. Thus, the CBR service class was created with the following set of features that help minimize latency and jitter to maintain excellent voice quality:

- CBR flows are buffered separately from each other and from flows in the CIR and BE classes. In contrast, the majority of other products mix up voice and data in the same buffer, so that voice packets get queued up behind data.
- CBR service flows are given strictly higher priority versus CIR and BE service flows, which implies that the system serves CIR and BE packets only after it has finished transmitting all the outstanding CBR packets.
- In the upstream, the system uses an optimized mode for CBR, known as Unsolicited Grants mechanism. This mechanism bypasses the normal request-grant mechanism for upstream traffic by having the BSU give automatic grants to a CBR flow.

In section 6.2.4 we describe in detail how CBR service flows can be configured to support voice traffic.

6 Configuring ServiceQ Parameters

Please refer to the Chapter 6 of the *PacketWave 1000 Series Base Station Equipment Installation, Configuration, and Operation Manual* for an overview on the use of the Configuration Manager (CM).

The two main decisions for configuring a Subscriber Unit are the classes of service for its service flows and peak rate data shaping. The main decision for configuring the Base Station Unit is what percentage of the available bandwidth to allocate to the three classes.

6.1 BSU Configuration Parameters

When configuring a wireless subsystem (WSS), the operator must determine the percentages of available bandwidth that should be available for each class of service: CBR, CIR, and BE.

The default allocation is 100% for BE. Obviously, this configuration will only be appropriate if no CIR or CBR flows are needed; in any other case it has to be modified. The percentages apply to the total upstream and total downstream bandwidth available, which depends on the frame and channel parameters of the WSS. Aperto Networks recommends the settings shown in for every supported channel width and the table shows the available DS and US bit rates for these cases. Any change in these parameters will result in a different available bit rate.

To configure these allocations, click on the **Channel** tab in the BSU CM and specify the percentage values for the "QoS Link Parameters" section, highlighted in yellow in Figure 6-1.

Table 6-1: Recommended TDD frame parameters and available bandwidth per channel width

Channel Width (MHz)	TDD Frame Size	DS Portion	US Portion	Number of REQs	Max Number of ACKs	DS Bandwidth (Mb/s)	US Bandwidth (Mb/s)
1	12500	8700	3720	2	4	1.081	0.300
1.75	10000	6980	2940	2	4	2.460	0.800
2	7500	5220	2200	2	4	2.640	0.800
3	7500	5220	2200	3	4	4.606	1.422
3.5	5000	3460	1460	2	4	5.482	1.655
4	5000	3460	1460	3	4	6.365	1.767
5	5000	3460	1460	3	4	8.017	2.559
6	5000	3460	1460	4	4	10.002	3.124
7	5000	3460	1460	4	4	10.002	3.124

If a service provider wishes to ensure all CIR and CBR guarantees will be met most of the time, the CIR and CBR percentages should be set so that they equal or exceed the total amount of CIR and CBR bandwidth expected to be used by simultaneously active service flows at any given time. For example, if the provider's statistical expectations of usage suggests that no more than 10 downstream 512 kb/s CIR flows will be active at the same time, this adds up to 5 Mb/s. If the total downstream bandwidth available is 8 Mb/s, then the CIR allocation should be at least 63%. Keep in mind that these 10 CIR flows may still use up more than 5 Mb/s of available CIR bandwidth if wireless link conditions require an SU to switch to more robust link parameters and slow down their link.

Refer back to sections 5.2 and 5.3 for more information on how to allocate bandwidth appropriately.

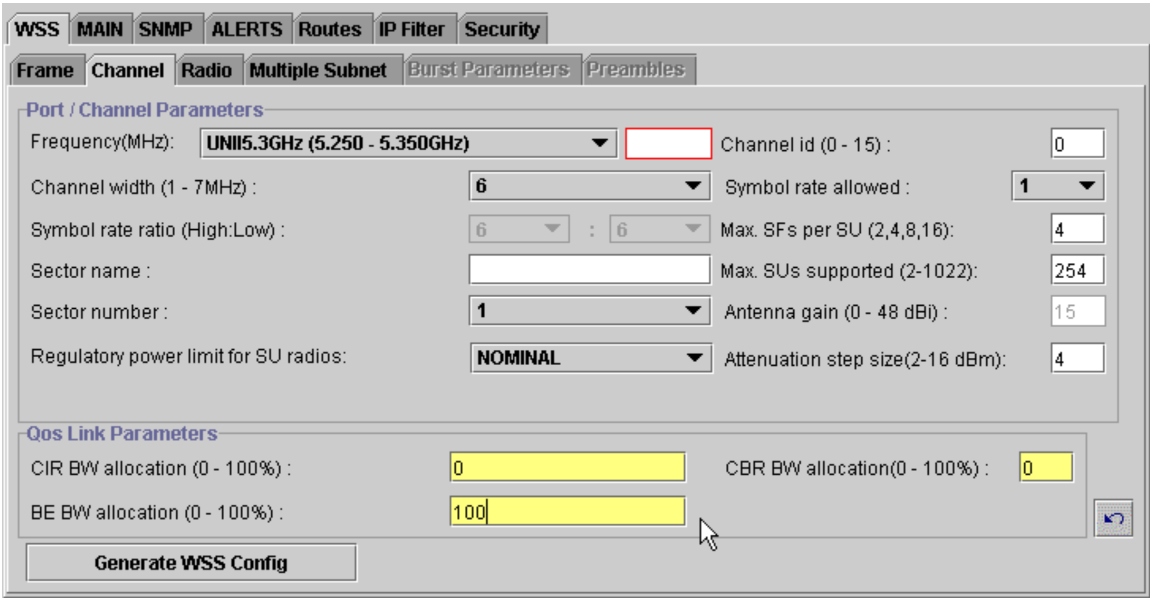


Figure 6-1: BSU QoS Link Parameters

6.2 SU Configuration Parameters

For each Service Flow, select the desired service class. For CIR, select the minimum reserved rate. For downstream CBR, select the bit rate. The sections that follow explain in detail what steps to follow in order to provide minimum bandwidth guarantees, peak-rate regulation or constant bit rate service.

PacketWave provides flexibility in how to offer different tiers of service to customers to maximize revenue. The two main choices are whether to guarantee a minimum bit rate (and how much), and whether to enforce a maximum bit rate (and again, how much). These choices need to be made before generating the SU's configuration file via the SU Configuration Manager (CM). To configure QoS, select the **Service Flow** tab in the SU CM.

6.2.1 Using a CIR Service Flow to specify a minimum guaranteed bit rate

To provide a user no guarantees, you do not need to change the service flow configuration, as Best Effort (BE) service is the default. To provide a minimum bit rate guarantee, you must create a new service flow, as the default BE flow cannot be configured to provide this guarantee.

6.2.1.1 Creating and Saving Service Flows

Select the **Service Flow** tab in the CM. Before you can create a new service flow, you must first save the default BE service flow by clicking the **Save** button, which is the floppy disk icon located at the bottom right of the CM window, as shown in Figure 6-2. This will save both the upstream and downstream configuration for the currently configured service flow. In general, you cannot create a new flow until you save the one being configured.

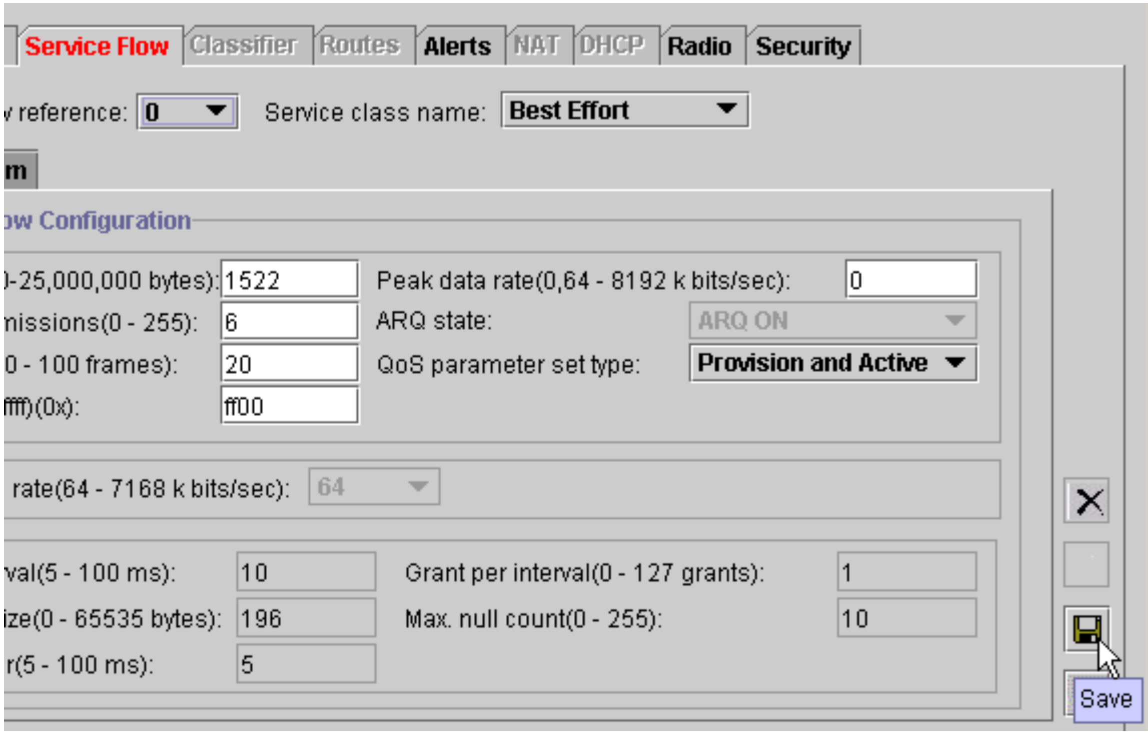
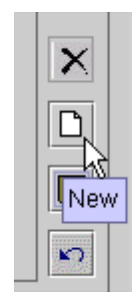


Figure 6-2: Saving the Default Service Flow



Once you hit the save button, the **New** button should appear right above it, as shown at right. Click this **New** button to create a new service flow for which you can guarantee a bit rate. If you ever create a service flow by accident, you can delete a created service flow by hitting the **Delete** button, which is above the **New** button. You will lose the entire configuration for the service flow you are currently configuring, so make sure this service flow is the one you want to delete! Note that the **Save**, **New**, and **Delete** buttons can also be similarly used for other configurable items, like classifiers and subnets.

The **Configured service flow reference** keeps track of which service flow you are configuring. The default best effort flow always has a reference number of 0. You can select a different flow to configure by selecting from the drop-down list, as shown in Figure 6-3. However, note that when you create a new service flow, CM will automatically switch to configure that flow, which will be reflected by the **Configured service flow reference** changing to the new flow's number. Therefore, you normally shouldn't ever have to select from this list unless you wish to go back to change a previously configured service flow.

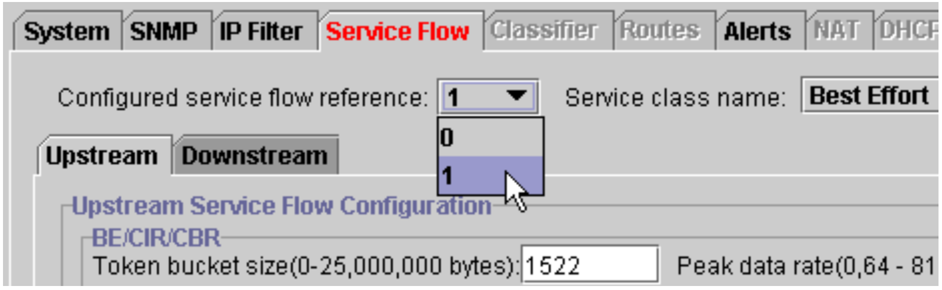


Figure 6-3: Configured service flow reference

6.2.1.2 Configuring a CIR Service Flow

To configure the new service flow to provide a bitrate guarantee, select CIR from the **Service class name** pull down menu (located on the top of the service flow window) shown in Figure 6-4 below.

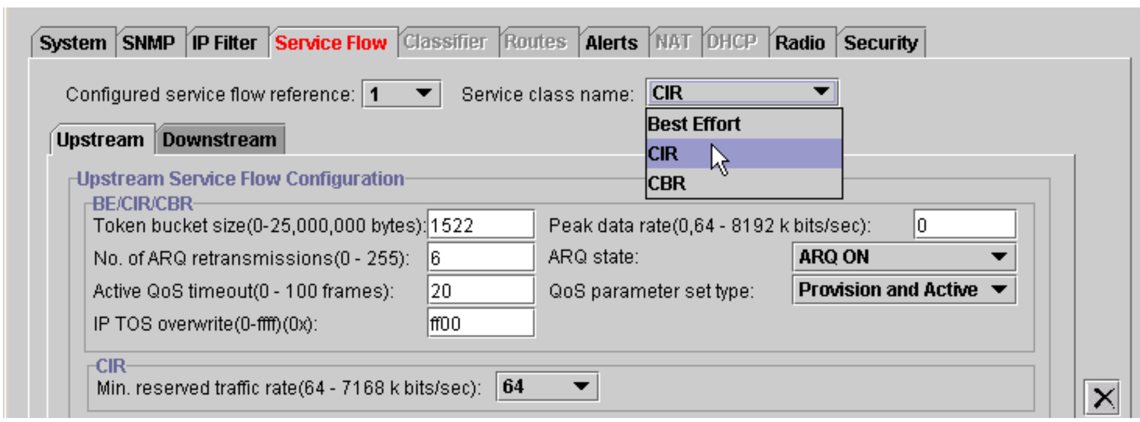


Figure 6-4: Configuring CIR

Next, specify the desired guaranteed traffic rate from the **Min. reserved traffic rate** drop down list (remember it's specified in kilobits per second, so for 1 Mb/s, select 1024).

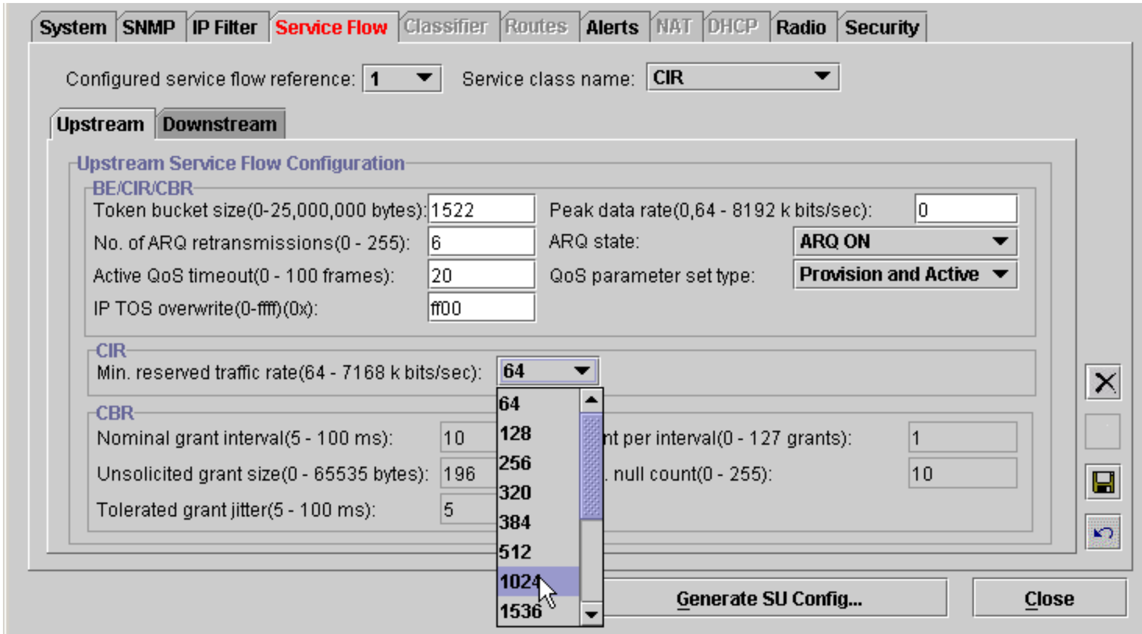


Figure 6-5: Configuring Upstream Minimum Reserved Rate

Hit the **Save** button to save these changes to the new service flow and move to the **Downstream** tab. Again, specify the desired **Min. reserved traffic rate** and hit the **Save** button, as shown in below (note the layout is different than for upstream).

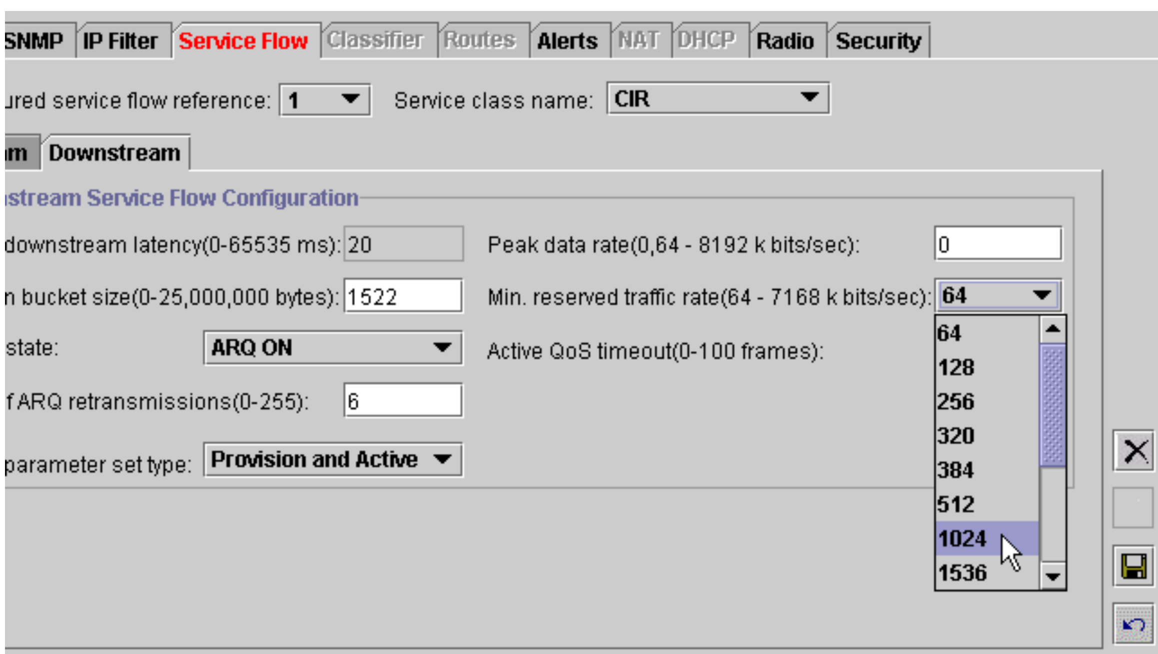


Figure 6-6: Configuring Downstream Minimum Reserved Rate

Note that the **Service Flow** tab turns red when there are changes that have not yet been saved, and returns to black once you hit the **Save** button. Also, note that the

Classifier tab turns red once you've created a new service flow. This is because for each service flow you create (not including the default service flow), you must create at least one classifier.

6.2.1.3 Configuring Classifiers

To create a classifier, move to the **Classifier** tab. Make sure the classifier's **Service flow reference** matches the CIR Service Flow's **Configured service flow reference** **(**refer to Figure 6-3), since that's the service flow where we want all traffic to go. If it does not, select the appropriate service flow from the **Service flow reference** drop-down list as shown in Figure 6-7. Note the default BE flow will not be available in this list, since it cannot have any classifiers, but instead receives traffic that is not matched by any classifier.

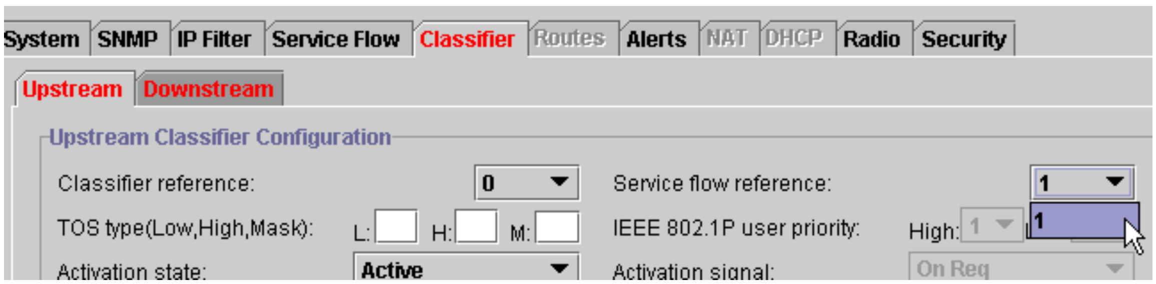


Figure 6-7: Selecting the Classifier's Service Flow

Since we want all traffic for this user to go to the newly created CIR flow, specify all 0's in both the **Destination MAC address** and **Destination MAC mask**, highlighted in yellow in Figure 6-8. This means every packet's destination MAC address will be masked to become all 0's before being compared to the classifier's destination MAC address of all 0's, so this rule will match all traffic.

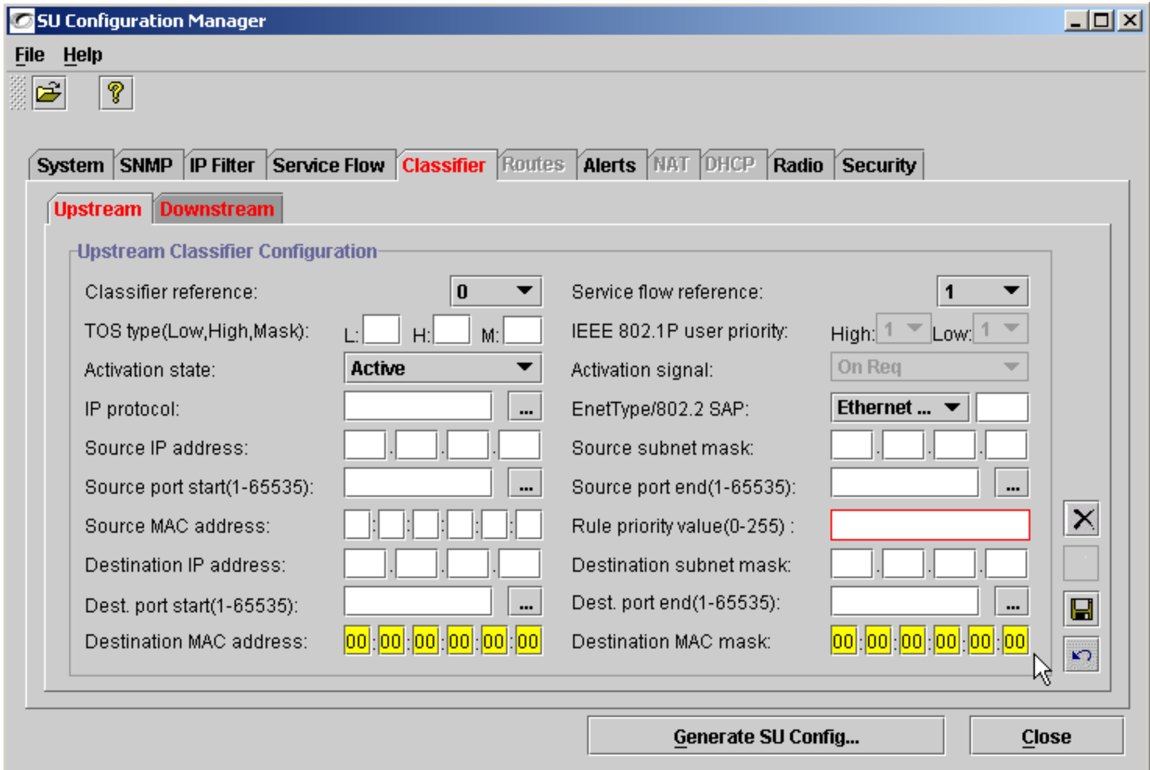


Figure 6-8: Specifying a Classifier to Match All Traffic

Next, you need to specify the **Rule priority value** for this classifier. The higher the priority number, the earlier the classifier rule will be checked. If you only have one classifier in this direction for this SU (and if you want all traffic to go to a single flow, you will have only one), the priority isn't important, so you can just enter 0 here, as shown in Figure 6-9. Setting this classifier at the lowest priority will also allow you to add higher priority classifiers to match a subset of the traffic for another service flow, and then allow all the remaining traffic to use this service flow. For example, you may want to add an additional service flow for voice.

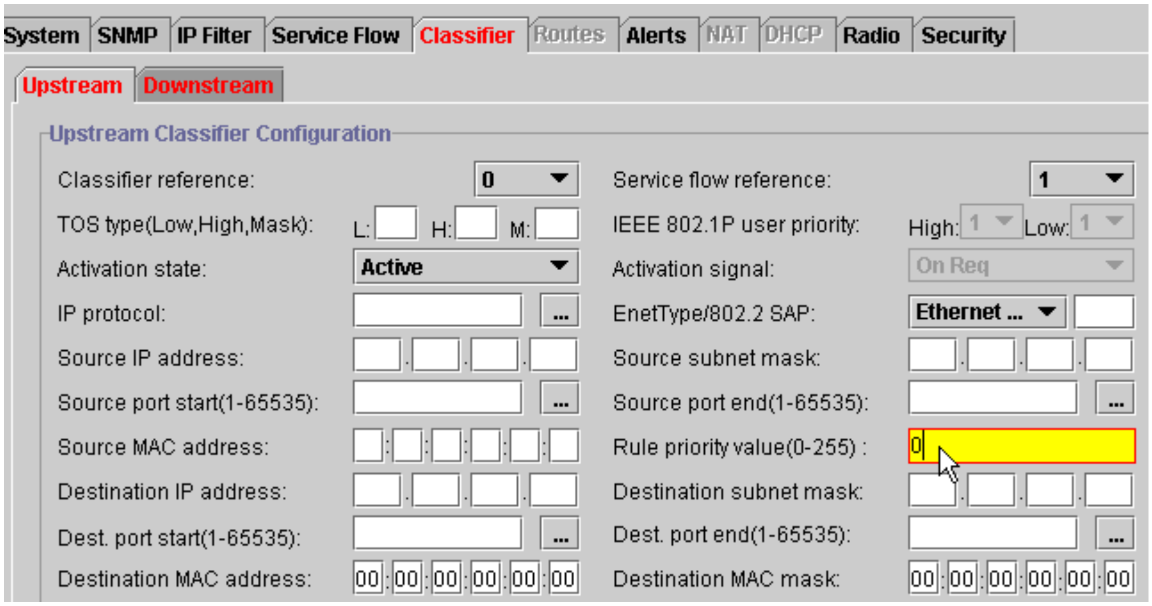


Figure 6-9: Classifier Priority

When you are finished, hit the **Save** button for the classifier, just as you did when you configured the service flow. Note, however, that this will save only the upstream or the downstream classifier, depending on which one you are currently configuring (unlike the **Save** button for service flow, which saves both at once). So if you have just configured a classifier for the upstream service flow when you hit the **Save** button, the **Upstream** classifier tab will return from red to black, but the **Downstream** classifier tab will still remain red since you have not yet configured a classifier for the downstream service flow you have created. Remember that for every upstream and downstream service flow you create (not including the default BE flows), you must create at least one classifier.

6.2.2 Using Traffic Shaping to specify a maximum allowed bit rate

To enforce a maximum bit rate (regardless of whether or not any minimum is guaranteed), go to the **Service Flow** tab and select the **Configured service flow reference** of the desired service flow for which you want a maximum bitrate (see Figure 6-3). Note that you *can* configure a maximum bit rate for the default BE service flow, which would be reference number 0.

Then select either the **Upstream** or **Downstream** tab, as desired (if you want to configure it for both, perform the following instructions for first one, then the other). The two parameters associated with traffic shaping are **Peak data rate** and **Token bucket size**. If the **Peak data rate** is set to 0, then traffic shaping is disabled, which is the default. To turn on traffic shaping, set the **Peak data rate** to the desired maximum bit rate, in kilobits/second. To allow burstiness, increase the **Token bucket size** beyond the minimum of 1522 bytes.

The below screen shows the **Peak data rate** for the upstream service flow set to 128 kb/s.

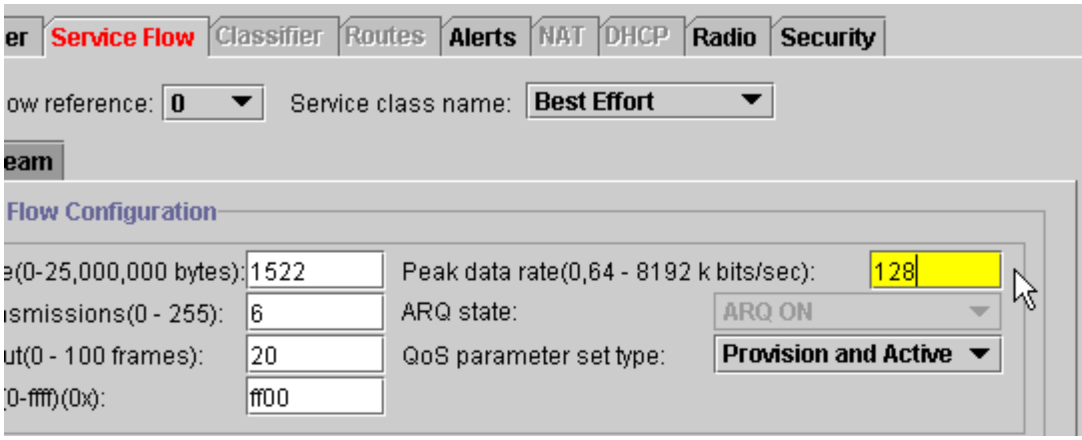


Figure 6-10: Configuring Traffic Shaping

Remember, shaping is configured independently for upstream and downstream. You may configure them identically, or you may constrain bandwidth more in one direction than the other, or have shaping enabled only in one direction. Also remember to hit the **Save** button after you are finished configuring both the upstream and downstream flows.

6.2.3 Advanced Parameters

Some of the configurable parameters should not need to be changed for the vast majority of cases.

Active QoS timeout is used in the admission control logic to determine how long an admitted CIR or CBR flow keeps their bandwidth reservation after they stop sending traffic. Refer to section 5.3 for more information on this.

ARQ state and **No. of ARQ retransmissions** controls whether to use the automatic retry mechanism. Refer to the *Aperto Networks White Paper on WMAC Parameters* for more information on this mechanism.

6.2.4 Voice Support

To add support for voice, you need to configure Constant Bit Rate (CBR) flows for each SU that needs voice support. Please refer to section 6.2.1.1 on how to save and add service flows. To configure the new service flow for CBR, refer to Figure 6-4: Configuring CIR, but select CBR instead of CIR. Alternatively, refer to the *PW1000 Installation, Configuration, and Operation Manual*.

6.2.4.1 Configuring Downstream CBR

Next, you need to provide the **Peak Data Rate** on the **Downstream** tab. Refer to section 6.2.2 for more details on how to do this. Refer to the formula in Table 6-3 to determine the appropriate **Peak Data Rate**, but in most cases, the minimum 64 kilobits/second should suffice. In both the **Upstream** and **Downstream** tabs, the **ARQ state** should be “ARQ Off, Return ACKs”, which should be the default for CBR service flows.

6.2.4.2 Configuring Upstream CBR

In addition, you will need to specify additional parameters in the CBR section on the **Upstream** tab which will allow you to take full advantage of CBR's capabilities.

PacketWave's RapidBurst technology requires a few extra parameters to reduce latency further for upstream CBR flows. These parameters, configurable in the **Upstream** tab, are shown highlighted in Figure 6-11.

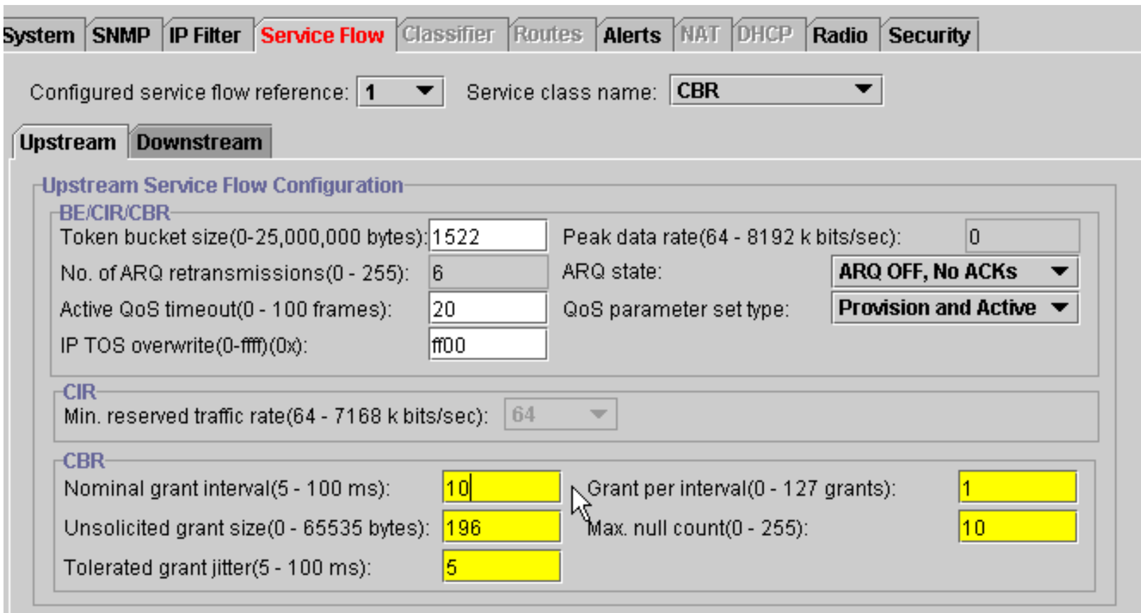


Figure 6-11: Configuring upstream CBR parameters

Nominal Grant Interval (5-100 ms): This parameter corresponds to the VoIP coder/decoder (codec) sampling rate (i.e., frame size). The default for G.711 and G.729 is 20 ms, 30 ms for G.723.1. The actually settings may differ from one VoIP device to another.

Grant per Interval (0-127 grants): This parameter corresponds to the number of simultaneous VoIP call sessions planned from the VoIP device (i.e., IAD) at the subscriber location. This number should not be greater than the number of voice interfaces on the IAD. Note that if a service flow is configured with more than 1 grant per interval, the scheduler will schedule the grants only when activity is detected (based on an internal mechanism).

Unsolicited Grant Size (0-65535 bytes): This parameter corresponds to the wireless payload for the VoIP codec selected. Use the numbers in the table below. The table also contains each codec's Peak Data Rate parameter to be entered in the DS tab.

Maximum null count: This is an advanced parameter that should not be changed. It controls how long the system will continue to provide unsolicited grants after the service flow has gone idle.

Tolerated grant jitter: This feature is currently not supported.

Table 6-3: CBR settings for voice support

Codec	Unsolicited Grant Size	DS Peak Data Rate (Kbps)
G.711	218 bytes	87 x (Grant per Interval)
G.729A	78 bytes	31 x (Grant per Interval)
G.723.1 (6.3 Kbps)	78 bytes	22 x (Grant per Interval)
G.723.1 (5.3 Kbps)	82 bytes	21 x (Grant per Interval)

6.2.4.3 Configuring a Classifier for an IP Address

One easy way to ensure voice traffic goes over the CBR service flow is to configure the classifier for the IP address of the VoIP device. To do this, select the **Classifier** tab and select the **Service flow reference** of the CBR flow (refer to Figure 6-7). Then, for the **Upstream** classifier, specify the IP address of the VoIP device in the **Source IP address** field, and specify 255.255.255.255 for the **Source subnet mask** (this is really not a subnet mask, but a mask of what bits in the packet's IP address to look at and what bits to ignore). Figure 6-12 shows an example where the IP address of the VoIP device is 123.45.67.89 (be sure to replace this with the actual IP address of your VoIP device).

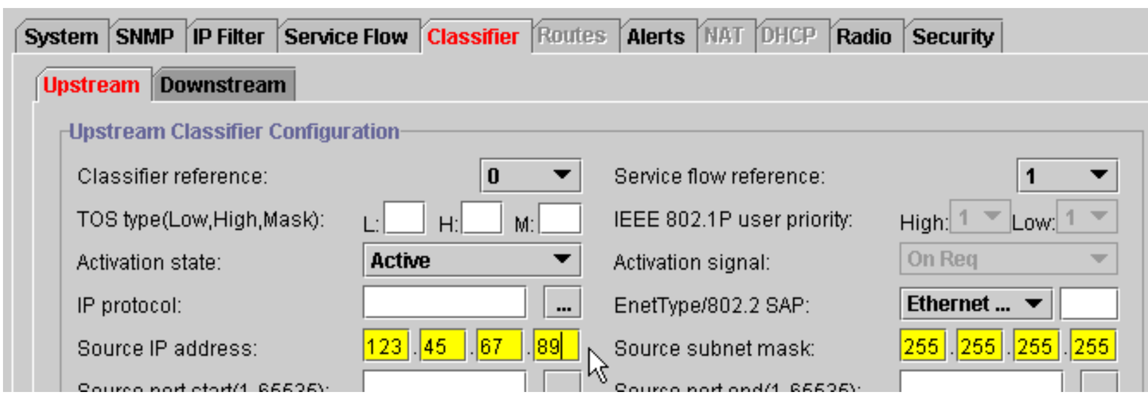


Figure 6-12: Source IP Classifier

If this is the only upstream classifier for this SU, then you can just put 0 for the **Rule priority value** (see Figure 6-9). If there are other classifiers, then choose how to prioritize them and give the higher priority classifiers a higher number in the **Rule priority value**. For example, if you have a classifier to match all traffic to go to a CIR service flow, you will want that classifier at a lower priority than the voice classifier (or else no traffic will go to the voice service flow). So you can set the voice classifier with a priority of 1, and the CIR classifier with a priority of 0.

When you are done with the upstream classifier, hit the **Save** button and move to the **Downstream** tab. Now specify the IP address of the VoIP device in the **Destination IP address** instead, as shown in Figure 6-13.

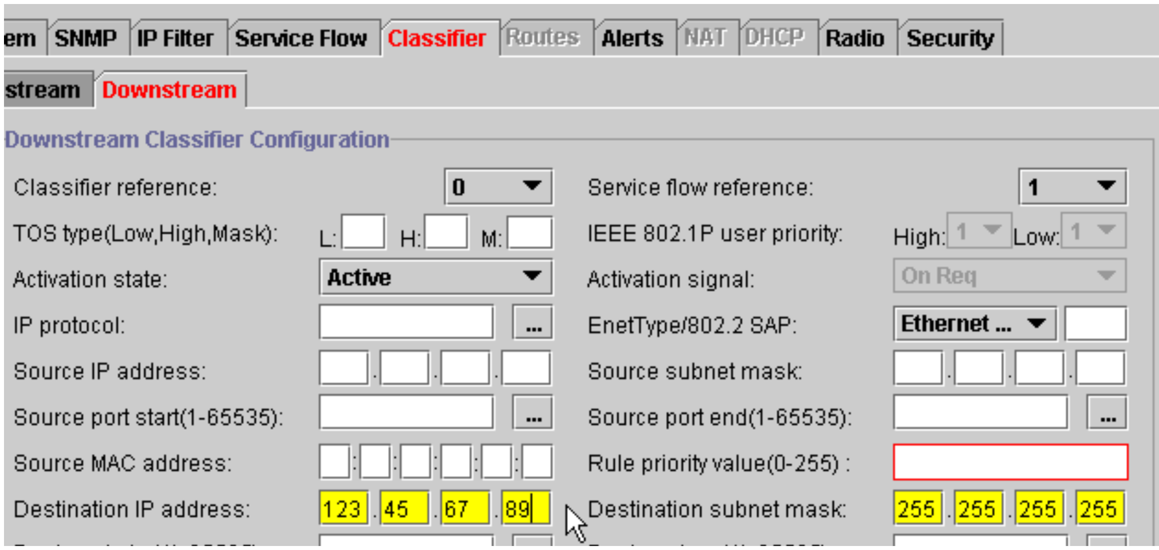


Figure 6-13: Destination IP Classifier

Again, specify the priority (see Figure 6-9) and hit the **Save** button. That's it!

7 Troubleshooting FAQ

Question 1 I have configured a CIR (or a CBR) service flow during my SU configuration, but I only see BE flows on the particular SU.

This question indicates that even though CIR and/or CBR service flows may have been properly configured on the SU, the BSU sector in which the SU is located does not have enough bandwidth reserved for CIR and/or CBR. The result is that the BSU rejects the registration request from the SU and the SU is provided with the default BE service flow.

This problem can be easily diagnosed through the Web GUI (BSU) by clicking on *Configuration* and selecting *Wireless Interface* and then *Channel.* The fix is to reconfigure the particular sector with an appropriate percentage of CIR or CBR and reboot the BSU.

Question 2 Incorrect Classifier rules result in traffic being served by the default BE service flow.

A quick way to check if a traffic stream is handled by the intended service flow is through the Web GUI. Open the BSU Web GUI and click on *Performance* and then on *BSU Flow Statistics.* If possible pause all traffic of the particular SU, mark down the *Packet Counters* (both downstream and upstream) and then start the traffic stream that you wish to be served by any flow x of that SU. You should see the packet counters of the particular service flow increase. If only the counters for the Flow with ID 0 increase, then the classifier rules are probably wrong and you need to revisit the SU configuration.

Question 3 My throughput is very low and the system is not being heavily used; what could the reason be?

Check whether the particular service flow has been configured with a peak rate.

Question 4 The throughput of a CIR service flow is below the minimum rate configured. What could the reason be?

This may be happening due to two reasons. One reason could be over-subscription in that particular sector. If the total of CIR service flows configured in that sector exceeds the CIR bandwidth allocated (through the BSU Configuration), some flows will be handled as BE whenever the CIR bandwidth is "fully" utilized. Another reason could be that the link modulation and/or coding for a particular SU may have changed to account for bad link conditions. In that case, admission control works seamlessly with the Adaptive Coding and Modulation scheme, such that the bandwidth guarantees continue to be met. For example, a 1 Mb/s CIR flow whose SU's link drops from 16QAM to QPSK will end up taking up 2 Mb/s worth of the available bandwidth, and the admission control algorithm will account for this accordingly. Note however, that if at the time of change the available CIR bandwidth is not sufficient to accommodate the 2 Mb/s, the service flow will be treated as BE.

Question 5 How is the *excess* available bandwidth shared among active CIR and BE flows?

As explained in section 5.1, the excess bandwidth is dynamically allocated in proportion to the weights of each active CIR flow and the weight of the BE group (provided at least one BE flow is active). Note that the excess bandwidth is defined as follows:

$$BW_{Excess} = BW_{Total} - \sum_{k \text{ is active}} CIR_k - BW_{BE}$$

where:

- CIR_k is the minimum CIR rate of flow k
- BW_{BE} is the bandwidth allocated to BE, provided at least one BE flow is active (otherwise it will be equal to zero)

Consider the following example: The aggregate upstream bandwidth in a given sector is 4 Mb/s, of which 1.5 Mb/s have been allocated to CIR and the remaining (2.5 Mb/s) to BE. Assume that at a given instance there in only one active CIR flow and several BE flows. The WFQ scheduler treats all the BE flows as a group, therefore it uses the following weights:

- the CIR service flow with minimum rate of 500 kbps (i.e. its WFQ weight is 0.5)
- the BE group (i.e. its WFQ weight is 2.5)

Assume also that all flows are bursty in that they require as much bandwidth as possible and no peak rate has been configured. The excess bandwidth in this case will be:

$$BW_{Excess} = BW_{Total} - \sum_{k \text{ is active}} CIR_k - BW_{BE} = 4 - 0.5 - 2.5 = 1 \text{ Mb/s}$$

Therefore the CIR flow, in addition to its guaranteed 500 kb/s, will see (0.5/3), or 16.7% of the extra 1 Mb/s whereas the BE flows will see a total of 2.5 Mb/s plus (2.5/3), or 83.3% of the 1 Mb/s.