White Paper

MPLS DiffServ-aware Traffic Engineering

Ina Minei
Software Engineer

Juniper Networks, Inc.
1194 North Mathilda Avenue
Sunnyvale, CA 94089 USA
408 745 2000 or 888 JUNIPER
www.juniper.net

Part Number: 200048-001

1. Executive Summary

Differentiated Services (DiffServ) enables scalable network designs with multiple classes of service. MPLS traffic engineering (TE) enables resource reservation, fault-tolerance, and optimization of transmission resources. MPLS DiffServ-TE combines the advantages of both DiffServ and TE. The result is the ability to give strict Quality of Service (QoS) guarantees while optimizing use of network resources. The QoS delivered by MPLS DiffServ-TE allows network operators to provide services that require strict performance guarantees such as voice and to consolidate IP and ATM/FR networks into a common core.

2. Introduction

In today's competitive market, service providers seek to increase their revenues while at the same time keeping capital and operational expenditures down. From a practical point of view, this means (1) providing new revenue-generating services, such as voice and guaranteed bandwidth for business-critical applications and (2) migrating Layer 2 services from legacy networks such as ATM and FR into the IP network infrastructure, thus eliminating the need to maintain several physical networks. The challenge lies in the fact that both the new and the legacy services usually require strict service level agreements (SLAs).

The SLAs define the service quality experienced by traffic transiting the network and are expressed in terms of latency, jitter, bandwidth guarantees, resilience in the face of failure, and downtime. The SLA requirements translate to two conditions: (1) different scheduling, queuing, and drop behavior based on the application type; and (2) bandwidth guarantees on a per-application basis.

To date, service providers have rolled out revenue-generating services in their networks using DiffServ alone. By assigning applications to different classes of service and marking the traffic appropriately, the first condition was met. However, to receive strict scheduling guarantees, it is not enough to mark traffic appropriately. If the traffic follows a path with inadequate resources to meet performance characteristics such as jitter or latency requirements, the SLAs cannot be met. In principle, service providers could solve this problem by using overprovisioning to avoid congestion altogether. Besides being wasteful with regards to resource utilization, this approach of "throwing bandwidth at the problem" cannot provide any guarantees when congestion is caused by link and/or node failures.

MPLS traffic engineering (MPLS-TE) sets up label-switched-paths (LSPs) along links with available resources, thus ensuring that bandwidth is always available for a particular flow and avoiding congestion both in the steady state and in failure scenarios. Because LSPs are established only where resources are available, overprovisioning is not necessary. Further optimization of transmission resources is achieved by allowing LSPs not to follow the shortest path, if the available resources along the shortest path are not sufficient. An added benefit of MPLS is that built-in mechanisms such as link protection and fast reroute provide resilience in the face of failure. The catch is that MPLS-TE is oblivious of the class of service (CoS) classification, operating on the available bandwidth at an aggregate level across all classes.

MPLS DiffServ-TE makes MPLS-TE aware of CoS, allowing resource reservation with CoS granularity and providing the fault-tolerance properties of MPLS at a per-CoS level. By combining the functionalities of both DiffServ and TE, MPLS DiffServ-TE delivers the QoS guarantees to meet strict SLAs such as the ones required for voice, ATM, and Frame Relay.

Note that even if resources are reserved on a per-CoS basis, and even if traffic is properly marked to conform to the CoS appropriate for the application, the SLAs still cannot be guaranteed unless further mechanisms, such as policing and admission control, are set in place to ensure that the traffic stays within the limits assumed when the resource reservation was made.

Section 3 of this paper introduces DiffServ and TE, presenting a few application scenarios and discussing how each of these technologies alone cannot provide a satisfactory solution. Section 4 gives an in-depth view of the MPLS DiffServ-TE technology as defined in the IETF standards. Section 5 highlights some of the advantages of the JUNOS DiffServ-TE implementation.

3. DiffServ and TE – The Current Tools

This section introduces the two building blocks of MPLS DiffServ-TE, DiffServ and TE, and explains why it is necessary to combine them in order to guarantee QoS in practical application scenarios.

3.1 DiffServ

The initial efforts to provide quality of service (QoS) in IP networks were based on a per-application-flow model (IntServ), in which individual applications requested QoS guarantees directly from the network. The RSVP signaling protocol was used to distribute the requests to the nodes in the network, and state needed to be maintained for each flow at every hop along the way. With millions of flows traversing IP networks, this approach proved to be unscalable and overly complex, and a more "coarse-grained" model was developed in the form of DiffServ.

DiffServ approaches the problem of QoS by dividing traffic into a small number of classes and allocating network resources on a per-class basis. To avoid the need for a signaling protocol, the class is marked directly on the packet, in the 6-bit DiffServ Code Point (DSCP) field. The DSCP field is part of the original type of service (ToS) field in the IP header. The IETF redefined the meaning of the little-used ToS field, splitting it into the 6-bit DSCP field and a 2-bit Explicit Congestion Notification (ECN) field, as shown in figure 1.

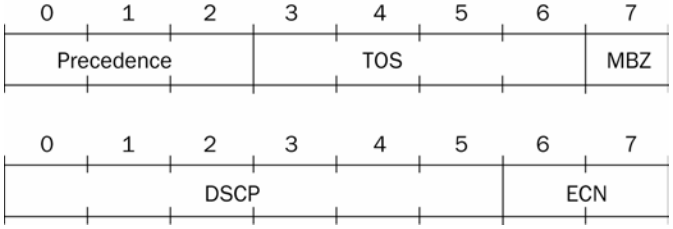


Figure 1: ToS and DSCP + ECN

The DSCP determines the QoS behavior of a packet at a particular node in the network. This is called the per-hop behavior (PHB) and is expressed in terms of the scheduling and drop preference that a packet experiences. From an implementation point of view, the PHB translates to the packet queue used for forwarding, the drop probability in case the queue exceeds a certain limit, the resources (buffers and bandwidth) allocated to each queue, and the frequency at which a queue is serviced. The IETF defined a set of 14 standard PHBs as follows:

- Best effort (BE). Traffic receives no special treatment.
- Expedited forwarding (EF). Traffic encounters minimal delay and low loss. From a practical point of view, this means a queue dedicated to EF traffic for which the arrival rate of packets is less than the service rate, so delay, jitter and loss due to congestion is unlikely. Voice and video streams are typical examples of traffic mapped to EF: they have constant rates and require minimal delay and loss.
- Twelve assured forwarding (AF) PHBs. Each PHB is defined by a queue number and a drop preference. The IETF recommends using four different queues with three levels of preference each, yielding a total of twelve distinct AF PHBs. The convention for naming the AF PHBs is AF*xy*, where *x* is the queue number and *y* is the level of drop preference. Thus, all packets from AF*1y* will be put in the same queue for forwarding, ensuring that packets from a single application cannot be reordered if they differ only in the drop preference. The AF PHBs are applicable for traffic that requires rate assurance but that does not require bounds on delay or jitter.

Although the IETF defined recommended DSCP values for each of the standard PHBs, vendors allow network operators to redefine the mapping between the DSCP and the PHB, and to define non standard PHBs. The important thing to keep in mind is that once the packet is marked with a particular DSCP value, its QoS treatment is defined at each hop it crosses. Thus, to ensure consistent QoS behavior, it is imperative to maintain consistent DSCP-to-PHB mappings. This requirement brings us to the notion of a DiffServ domain, which is a set of DiffServ capable nodes with 1) a common set of defined PHBs, 2) the same DSCP-to-PHB mappings, and 3) a unified service provisioning policy. Usually a DiffServ domain operates under a single administrative authority. At the edge of a DiffServ domain, the traffic is marked with the DSCP values that yield the desired per hop behavior and ultimately the desired QoS.

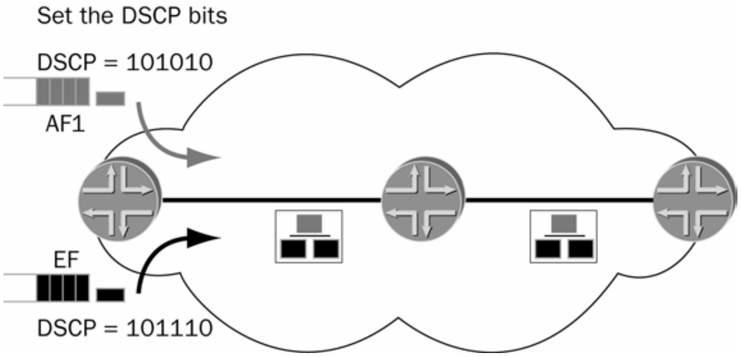


Figure 2: A DiffServ domain. Voice traffic is marked at the entrance to the domain with EF, data traffic with BE. Every hop examines the DSCP bits and maps the traffic to a different queue.

To summarize, DiffServ provides differential forwarding treatment to traffic, thus enforcing QoS for different traffic flows. It is a scalable solution that does not require per-flow signaling and state maintenance in the core. However, it cannot guarantee QoS if the path followed by the traffic does not have adequate resources to meet the QoS requirements.

3.2 MPLS DiffServ

RFC 3270 describes the mechanisms for MPLS support of DiffServ. The first challenge with supporting DiffServ in an MPLS network is that label-switching routers (LSRs) make their forwarding decisions based on the MPLS shim header alone, so the PHB needs to be inferred from it. The IETF solved this problem by assigning the three experimental (EXP) bits in the MPLS header to carry DiffServ information in MPLS (see Figure 3).

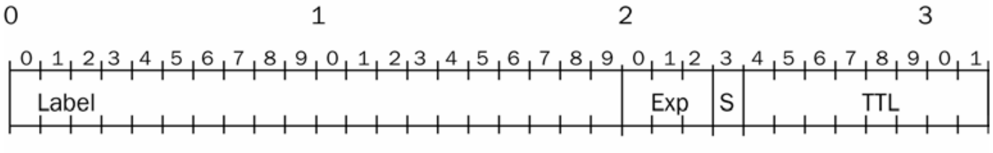


Figure 3: The MPLS header

This solution solves the initial problem of conveying the desired PHB in the MPLS header, while introducing a new one: how does one map DSCP values expressed in a 6-bit field that can encode up to 64 values, into a 3-bit EXP field that can carry at most eight distinct values? There are two solutions to this problem, discussed separately below.

The first solution applies to networks that support less than eight PHBs. Here, the mapping is straightforward: a particular DSCP is equivalent to a particular EXP combination and maps to a particular PHB (scheduling and drop priority). During forwarding, the label determines where to forward the packet, and the EXP bits determine the PHB. The EXP bits are not a property that is signaled when the label-switched path (LSP) is established, but rather they are a value that is configured. The EXP bits can be set according to the DSCP bits of the IP packets carried in the LSP, or they can be set by the network operator. LSPs for which the PHB is inferred from the EXP bits are called E-LSPs (where E stands for "EXP-inferred"). E-LSPs can carry packets with up to eight distinct per-hop behaviors in a single LSP. See Figure 4.

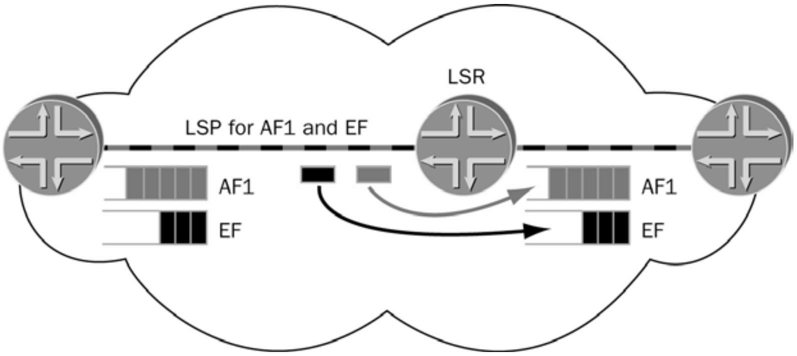


Figure 4: An E-LSP can carry traffic from Multiple PHBs.

The second solution applies to networks that support more than eight PHBs. Here, the EXP bits alone cannot carry all the necessary information to distinguish between PHBs. The only other field in the MPLS header that can be used for this purpose is the label itself. During forwarding, the label determines where to forward the packet and what scheduling behavior to grant it, and the EXP bits convey information regarding the drop priority assigned to a packet. Thus, the PHB is determined from both the label and the EXP bits. Because the label is implicitly tied to a per-hop-behavior, this information needs to be conveyed when the LSP is signaled. LSPs which use the label to convey information about the desired PHB are called L-LSPs (where L stands for "label-inferred"). L-LSPs can carry packets from a single PHB, or from several PHBs that have the same scheduling regimen but differ in their drop priorities (such as AF*xy* where *x* is constant and *y* is not constant). See Figure 5.

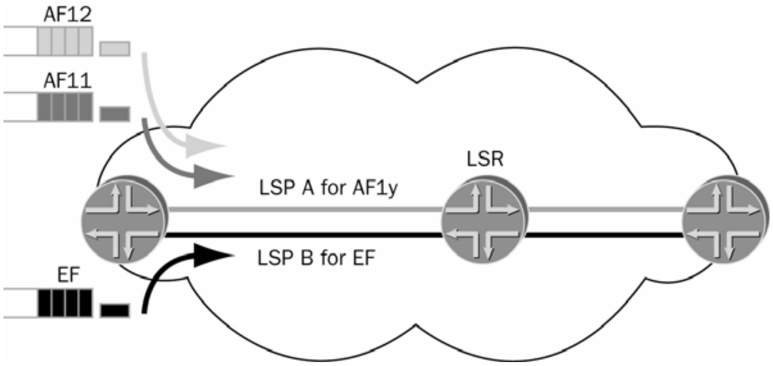


Figure 5: An L-LSP can carry traffic from a single PHB or from several PHBs that share the same scheduling behavior. LSP A carries traffic from AF11 and AF12.

The following table summarizes the differences between E-LSPs and L-LSPs:

E-LSP	**L-LSP**
PHB is determined by the EXP bits	PHB is determined by the label or by the label and EXP bits together
Can carry traffic with up to 8 distinct PHBs in a single LSP	A single PHB per LSP or several PHBs with the same scheduling regimen and different drop priorities
Conservative label usage and state maintenance, because the label is used only for conveying path information	Uses more labels and keeps more state, because the label conveys information about both the path and the scheduling behavior
No signaling is required to convey the PHB information	The PHB information needs to be signaled when the LSP is established
Up to 8 PHBs can be supported in the network when only E-LSPs are used. E-LSPs can be used in conjunction with L-LSPs when more PHBs are required	Any number of PHBs can be supported in the network

As always, the answer to the question of which type of LSP is better depends on the particular application scenario.

3.3 Traffic Engineering with MPLS (MPLS-TE)

Traffic engineering is used to achieve performance objectives such as optimization of network resources and placement of traffic on particular links. From a practical point of view, this means computing a path from source to destination that is subject to a set of constraints, and forwarding traffic along this path. Forwarding traffic along such a path is not possible with IP, since the IP forwarding decision is made independently at each hop, and is based solely on the packet's IP destination address.
MPLS can easily achieve forwarding traffic along an arbitrary path. The explicit routing capabilities of MPLS allow the originator of the LSP to do the path computation, establish MPLS forwarding state along the path, and map packets into that LSP. Once a packet is mapped onto an LSP, forwarding is done based on the label, and none of the intermediate hops makes any independent forwarding decisions based on the packet's IP destination.

MPLS-TE introduces the concept of LSP priorities. The purpose of priorities is to mark some LSPs as more important than others and allow them to confiscate resources from less important LSPs (preempt the less important LSPs). Doing this guarantees that 1) in the absence of important LSPs, resources may be reserved by less important LSPs, 2) an important LSP always establishes along the most optimal (shortest) path, regardless of existing reservations and 3) when LSPs need to reroute (e.g., following a link failure) important LSPs have a better chance of finding an alternate path. MPLS-TE defines eight priority levels, with 0 as the best and 7 as the worst value. An LSP has two priorities associated with it: a setup priority and a hold priority. The setup priority controls access to the resources at the time of LSP establishment, and the hold priority controls access to the resources for an LSP that is already established. At LSP setup time, if insufficient resources are available, the setup priority of the new LSP is compared to the hold priority of the LSPs using the resources, in order to determine if the new LSP can preempt any of the existing LSPs and take over their resources.

As mentioned above, the goal of traffic engineering is to find a path in the network that meets a series of constraints. Thus, these constraints need to be taken into account when calculating feasible paths to a destination. Some of these constraints are 1) the bandwidth requested for a particular LSP (for example, 10Mbs from source *x* to destination *y*), 2) the administrative attributes ("colors") of the links that the traffic is allowed to cross (for example, no low-latency links, where low-latency links are marked with a particular administrative attribute), 3) the number of hops that the traffic is allowed to cross , 4) the priority of this LSP when compared to other LSPs (for example, one out of eight possible priority levels). Other constraints are also possible.
Calculating a path that satisfies these constraints requires that the information about whether the constraints can be met is available for each link, and this information be distributed to all the nodes that perform path calculation. This means that the relevant

link properties have to be advertised throughout the network. This is achieved by adding TE-specific extensions to the link-state protocols IS-IS and OSPF that allow them to advertise not just the state (up/down) of the links, but also the link's administrative attributes and the bandwidth that is available for use by trunks at each of the eight priority levels. In this way, each node has knowledge of the current properties of all the links in the network.

Once this information is available, a modified version of the shortest-path-first (SPF) algorithm, called constrained SPF (CSPF), can be used by the ingress node to calculate a path that complies with the given constraints. Conceptually, CSPF operates in the same way as SPF, except it first prunes from the topology all links that do not satisfy the constraints. For example, if the constraint is bandwidth, CSPF prunes from the topology links that don't have enough bandwidth. Figure 6 shows a network topology and two LSPs with bandwidth requirements. Once the LSP A-C is set up, no resources for the LSP B-C are available along the shortest path, the links on the shortest path are pruned from the topology and CSPF picks the alternate path as the best available.

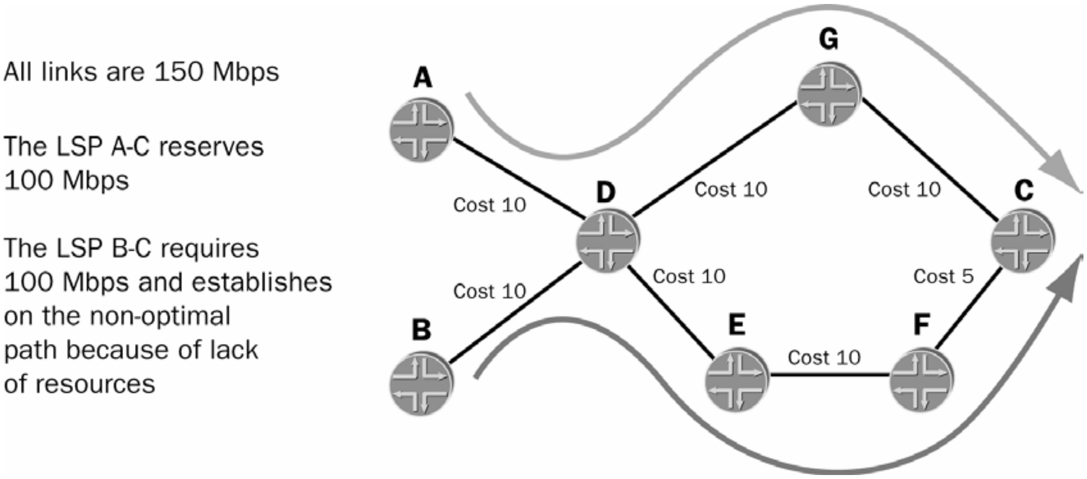

All links are 150 Mbps
The LSP A-C reserves 100 Mbps
The LSP B-C requires 100 Mbps and establishes on the non-optimal path because of lack of resources
A
B
D
G
C
E
F
Cost 10
Cost 10
Cost 10
Cost 10
Cost 10
Cost 10
Cost 5

Figure 6: An LSP can take a path other than the shortest path when resources are not available.

Finally, after a path has been successfully calculated, MPLS forwarding state is established along that path by using RSVP-TE as a label distribution protocol. As the path is set up, the available resources are updated at each node and the other nodes are informed of the changes through the IGP.

So far, we have seen how MPLS-TE can be used to build paths with bandwidth guarantees and how paths can be made to avoid certain links by marking such links with the appropriate administrative value and excluding them from the path computation. The remaining challenge is that MPLS-TE operates at the aggregate level across all the

DiffServ classes of service and as a result it cannot give bandwidth guarantees on a per class basis.

3.4 Application Scenarios

In RFC 3564, a few application scenarios are presented that cannot be solved using DiffServ or TE alone. These scenarios form the basis for the requirements that led to the development of the DiffServ-TE solution in the IETF, and are presented in this section.

3.4.1 Limiting the Proportion of Traffic from a Particular Class on a Link

The first scenario involves a network with two types of traffic: voice and data. The goal is to maintain good quality for the voice traffic, which in practical terms means low jitter, delay, and loss, while at the same time servicing the data traffic. The DiffServ solution for this scenario is to map the voice traffic to a PHB that guarantees low delay and loss, such as EF.

The problem is that DiffServ alone cannot give the required guarantees for the following reason: The delay encountered by the voice traffic is the sum of the propagation delay experienced by the packet as it traverses the network and of the queuing and transmission delays incurred at each hop. The propagation and transmission delays are effectively constant, so to enforce a small jitter on the delay, the queuing delay must be minimized. A short queuing delay requires a short queue, which from a practical point of view means that only a limited proportion of the queue buffers can be used for voice traffic.

Thus, the requirement becomes “limit the proportion of voice traffic on each link.”
In the past, service providers used overprovisioning to achieve this goal, making sure that more bandwidth is available than will ever be necessary. However, overprovisioning has its own costs, and, while it may work well in the normal case, it can give no guarantees in the failure scenario. Figure 7 shows a network operating under such a regimen. Under normal conditions, the voice traffic takes the path A-C-D, which is the shortest path. The link capacity is large, so the percentage of the voice traffic on each link is acceptable. When the link C-D fails, the traffic reroutes on the next best path, A-C-G-D. The link C-G is low-capacity, and the percentage of voice traffic becomes too large. Instead, the traffic should have rerouted on the path A-C-E-F-D.

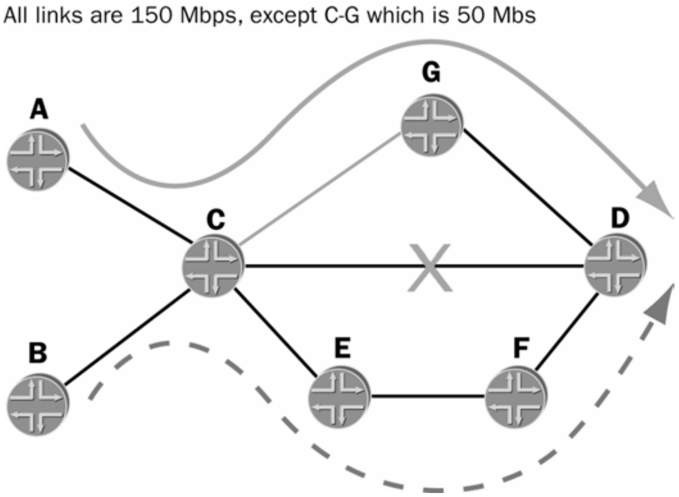


Figure 7: Overprovisioning cannot provide guarantees in a failure scenario.

Taking the solution a step further, limiting the proportion of voice traffic on all links can be achieved by artificially limiting the available bandwidth on a link to the proportion suitable to satisfy the voice traffic requirements alone and using TE to ensure that traffic (voice and data) is mapped in such a way as to honor these artificially lower available resources. This solution provides the requested functionality but wastes resources because bandwidth that could be allocated to delay-insensitive data traffic is now idle and unavailable. The root of the problem is that TE cannot distinguish between the two types of traffic and cannot enforce allocations at a per-traffic-type granularity.

3.4.2. Maintain Relative Proportions of Traffic on Links

The second scenario extends the previous example to a network that supports three traffic types that map to three "classes of service." The proportion of the different traffic types depends on the source and destination of the traffic. The challenge for the service provider is to configure the queue sizes and queue scheduling policies on each link to ensure that the correct PHB is given to each class. It is impractical to configure these parameters based on the link load at a given time: changes in routing, link/node failures, and preemption between LSPs make the link load a very dynamic property. Instead, from an operational and maintainability point of view it would be ideal to fix the relative proportions of each traffic type on the links, allocate the queue sizes and scheduling policies accordingly, and use TE to make the traffic comply with the available resources. This solution requires TE to enforce different bandwidth constraints for different classes of traffic.

3.4.3 Guaranteed Bandwidth Services

In this application, which is very similar to the first example, there are two types of traffic: best effort and "guaranteed bandwidth." The guaranteed bandwidth traffic must comply with a given SLA. The goal is to provide the required service level to the guaranteed traffic and also be able to traffic engineer the best-effort traffic. As in the first example, in order to enforce strict SLAs, the guaranteed bandwidth traffic must be engineered to take up only a percentage of the link, and TE must be employed to ensure this requirement. In addition, the best-effort traffic must also be traffic engineered. Here again, TE must have knowledge of the type of traffic.

4. DiffServ-TE - Leading-edge Tools

This section examines how per-traffic-type behavior is enforced both at forwarding time and at LSP setup time.

4.1 Class Types

The basic DiffServ-TE requirement is to be able to make separate bandwidth reservations for different classes of traffic. This implies keeping track of how much bandwidth is available for each type of traffic at any given time on all routers throughout the network.

For this purpose, RFC 3564 introduces the concept of a class type (CT) as follows:
"The set of traffic trunks crossing a link, that is governed by a specific set of bandwidth constraints. CT is used for the purposes of link bandwidth allocation, constraint based routing, and admission control. A given traffic trunk belongs to the same CT at all links."

RFC 3564 does not mandate a particular mapping of traffic to CTs, leaving this decision to the individual vendors. The JUNOS implementation maps traffic that shares the same scheduling behavior to the same CT. Thus, in JUNOS, one can think of a CT in terms of a queue and its associated resources. Because the PHB is defined by both the queue and the drop priority, a CT may carry traffic from more than one DiffServ class of service, assuming that they all map to the same scheduler queue (e.g., the AF2*x* PHBs).

The IETF requires support of up to eight CTs referred to as CT0 through CT7. LSPs that are traffic-engineered to guarantee bandwidth from a particular CT are referred to as DiffServ-TE LSPs. In the current IETF model, a DiffServ-TE LSP can only carry traffic from one CT. LSPs that transport traffic from the same CT can use the same or different preemption priorities. By convention, the best-effort traffic is mapped to CT0. Because all pre-DiffServ-TE LSPs are considered to be best effort, they are mapped to CT0.

Let us revisit the application scenario from Section 3.4.1 and discuss it in terms of CTs. The voice and data network in this example supports two DiffServ PHBs, EF and BE (for voice and data traffic, respectively). The goal is to provide service guarantees to the EF traffic. Two scheduler queues are configured on each link, one for BE and one for EF. CT0 is mapped to the BE queue and CT1 is mapped to the EF queue. The bandwidth available for CT1 (the voice traffic) is limited to the percentage of the link required to ensure small queuing delays for the voice traffic. Separate TE-LSPs are established with bandwidth requirements from CT0 and from CT1.

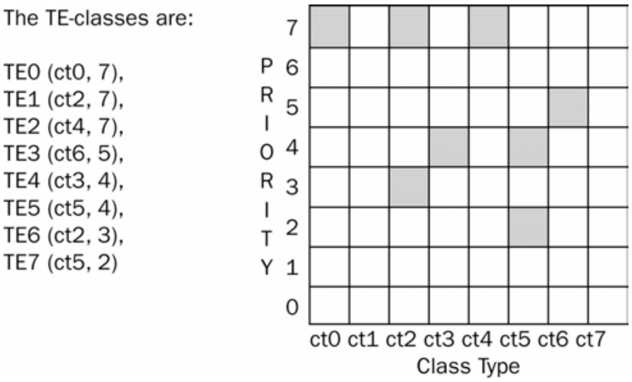
In the following sections, we look at how LSPs are established with per-CT bandwidth requirements.

4.2 Path Computation

In Section 3.3, we discussed how CSPF computes paths that comply with user-defined constraints such as bandwidth and link attributes. DiffServ-TE adds the available bandwidth for each of the eight CTs as a constraint that can be applied to a path. Therefore, CSPF is enhanced to take into account CT-specific bandwidth at a given priority as a constraint when computing a path. For the computation to succeed, the available bandwidth per-CT at all priority levels must be known for each link.

This means that the link-state IGPs must advertise the available bandwidth per-CT at each priority level on every link. Recall that there are eight CTs and eight priority levels, giving a total of 64 values that need to be carried by the link-state protocols. In an ideal world, all 64 values would be signaled and stored for each link. However, the IETF decided to limit the advertisements to eight values out of the possible 64.

For this purpose, a TE-class is defined as a combination of (CT, priority). DiffServ-TE supports a maximum of eight TE-classes, TE0 through TE7, which can be selected from the 64 possible CT-priority combinations via configuration. At one extreme, there is a single CT with eight priority levels, very much like the existing TE implementation. At the other extreme, there are eight distinct CTs, with a single priority level. Figure 8 shows the 64 combinations of class type and priority, and a choice of eight TE-classes.



The TE-classes are:

TE0 (ct0, 7),
TE1 (ct2, 7),
TE2 (ct4, 7),
TE3 (ct6, 5),
TE4 (ct3, 4),
TE5 (ct5, 4),
TE6 (ct2, 3),
TE7 (ct5, 2)

Figure 8: Picking eight TE-classes out of the 64 possible combinations

The link-state IGPs advertise the available bandwidth for each TE-class. [DSTE-PROTO] mandates that this advertisement be made using the existing Unreserved Bandwidth TLV, which was previously used to disseminate unreserved bandwidth for TE. Therefore, the information that is available to CSPF through the IGPs is relevant only for the CT and priority combinations that form valid TE-classes. Thus, in order for CSPF to perform a meaningful calculation, the CT and priority levels chosen for an LSP must correspond to one of the configured TE-classes.

To summarize, to compute a path with per-CT bandwidth constraints, CSPF is enhanced to handle per-CT reservation requirements, and the IGPs are enhanced to carry per-CT available bandwidth at different priority levels.

4.3 Path Signaling

After the path is calculated, it is signaled and admission control and bandwidth accounting are performed at each hop. [DSTE-PROTO] defines the necessary extensions to RSVP-TE that allow it to establish paths with per-CT bandwidth reservations.[1]
The CT information for an LSP is carried in the new Classtype object (CT object) in the RSVP path message, and specifies the CT from which the bandwidth reservation is requested. Two rules ensure that it is possible to deploy DiffServ-TE incrementally in the network: 1) the CT object is present only for LSPs from CT1 through CT7 (if the CT object is missing, CT0 is assumed), and 2) a node that receives a path message with the CT object and does not recognize it rejects the path establishment. These two rules ensure that establishment of LSPs with per-CT reservation is possible only through DiffServ-TE-

[1] Note that although CR-LDP also supports explicit routing, no extensions are defined for it because the IETF decided in RFC 3468 to abandon new development for CR-LDP.

aware nodes, while pre-DiffServ-TE LSPs which are considered to belong to CT0 can cross both old and new nodes.

The CT information carried in the path message specifies the CT over which admission control is performed at each node along the path. If a node along the path determines that enough resources are available and the new LSP is accepted, the node performs bandwidth accounting and calculates the new available bandwidth per-CT and priority level. This information is then fed back into the IGPs.

To summarize, for each LSP, the CT is implicitly signaled for CT0 and explicitly signaled for all other CTs. The CT is necessary to perform the calculation of the available resources. But how is this calculation performed?

4.4 Bandwidth Constraint Models

One of the most important aspects of the available bandwidth calculation is the allocation of bandwidth among the different CTs. The percentage of the link's bandwidth that a CT (or a group of CTs) may take up is called a bandwidth constraint (BC). RFC 3564 defines the term "bandwidth constraint model" to denote the relationship between CTs and BCs.

4.4.1 The Maximum Allocation Model (MAM)

The most intuitive bandwidth constraint model maps one BC to one CT. This model is called the maximum allocation model (MAM) and is defined in [DSTE-MAM]. From a practical point of view, the link bandwidth is simply divided among the different CTs, as illustrated in Figure 9.

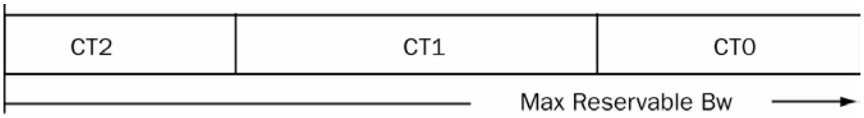


Figure 9: The allocation of bandwidth to CTs in the MAM model (for simplicity, only 3 CTs are shown.)

The problem with MAM is that because it is not possible to share unused bandwidth between CTs, bandwidth may be wasted instead of being used for carrying other CTs. Consider the network shown in Figure 10. In the absence of voice LSPs, bandwidth is available on all the links on the shortest path, but this bandwidth cannot be used for setting up another data LSP. The second data LSP is forced to follow a non-optimal path, even though bandwidth is available on the shortest path. On the other hand, after both data LSPs have been set up, if a voice LSP needs to be established, bandwidth is available for it on the shortest path.

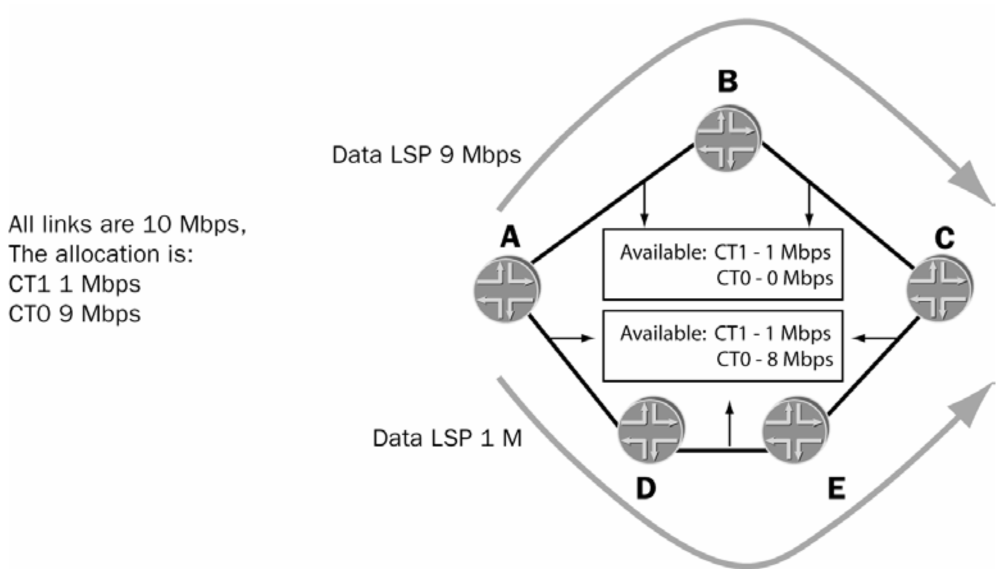


Figure 10: Even if no voice LSPs are established, the bandwidth allocated for voice cannot be used to carry data.

The benefit of MAM is that it achieves complete isolation between different CTs; thus, no priorities need to be configured between LSPs carrying traffic from different CTs. In the network shown in Figure 10, when the operator wants to set up a voice LSP, the resources are guaranteed to be available and no preemption of data LSPs is necessary, or indeed possible.

The available bandwidth for the MAM model is accounted in a similar way as for TE, except it is done on a per-CT basis. To calculate the bandwidth available for CT*n* at priority *m*, subtract from the bandwidth allocated to CT*n* the sum of the bandwidths allocated for LSPs of CT*n* at all priority levels that are better or equal to *m*.

4.4.1 The Russian Dolls Model (RDM)

The Russian dolls bandwidth allocation model (RDM) defined in [DSTE-RDM] improves bandwidth efficiency over the MAM model by allowing CTs to share bandwidth. In this model, CT7 is the traffic with the strictest QoS requirements and CT0 is the best-effort traffic. The degree of sharing varies between two extremes. At one end of the spectrum, BC7 is a fixed percentage of the link bandwidth that is reserved for traffic from CT7 only. At the other end of the spectrum, BC0 represents the entire link bandwidth and is shared among all CTs. Between these two extremes are various degrees of sharing: BC6 accommodates traffic from CT7 and CT6, BC5 from CT7, CT6 and CT5 and so on. This model is very much like the Russian doll toy, where one big doll (BC0) contains a smaller doll (BC1) which contains a yet smaller doll (BC2), and so on.

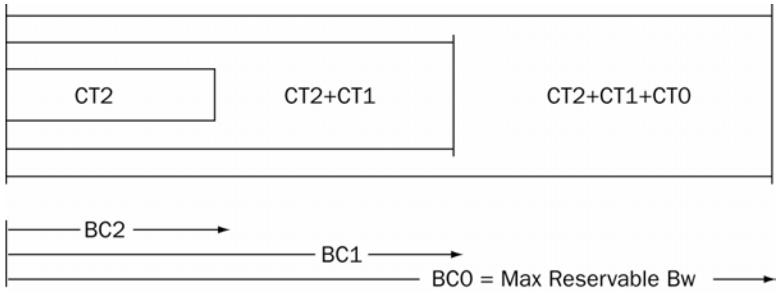


Figure 11: Russian dolls bandwidth allocation (for simplicity, only 3 CTs are shown)

The advantage of RDM relative to MAM is that it provides efficient bandwidth usage through sharing. Consider the network from Figure 12, which carries voice traffic and data traffic. The total bandwidth available on each link is 10Mbs. 1Mbs is allocated to BC1 and 10Mbs are allocated to BC0. What this means is that each link may carry between 0 and 1Mbs of voice traffic and use the rest for data. Assuming that a data LSP is already established over the path A-B-C, in the absence of voice traffic, a second data LSP can be established to take advantage of the unused bandwidth. Another useful property that is achieved through sharing is cheap overprovisioning for real-time traffic. Since the extra bandwidth can be used by other types of traffic, allocating it to the real-time class does not affect the overall throughput of the network.

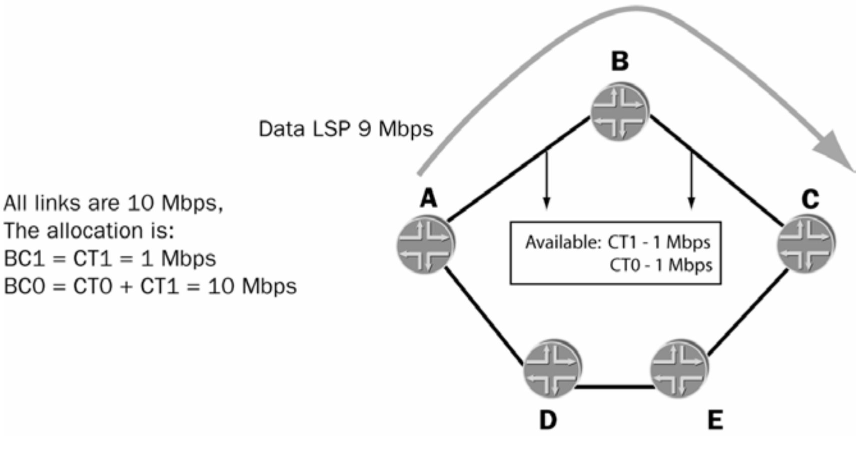


Figure 12: With the RDM model, in the absence of voice traffic, the available bandwidth can be used by data.

The disadvantage of RDM relative to MAM is that there is no isolation between the different CTs, and preemption must be used to ensure that each CT is guaranteed its share of bandwidth no matter the level of contention by other CTs. In the network shown in Figure 12, if after establishing the second data LSPs the operator wants to establish a voice LSP, he will find that no resources are available for the voice traffic, as shown in figure 13a. Thus, one of the data LSPs must be preempted: otherwise, bandwidth is not guaranteed for the voice traffic. This means that voice and data LSPs must be given different priorities, because they share bandwidth resources. Figure 13b shows the same network, but with voice LSPs at priority 0 and data LSPs at priority 1. (Recall that the best priority is priority 0 and the worst priority is priority 7). When the voice LSP is established, it preempts one of the data LSPs. Note that a voice LSP can only preempt the data LSP if the voice LSP's bandwidth requirement is such that the CT1 allocation on the link is not exceeded.

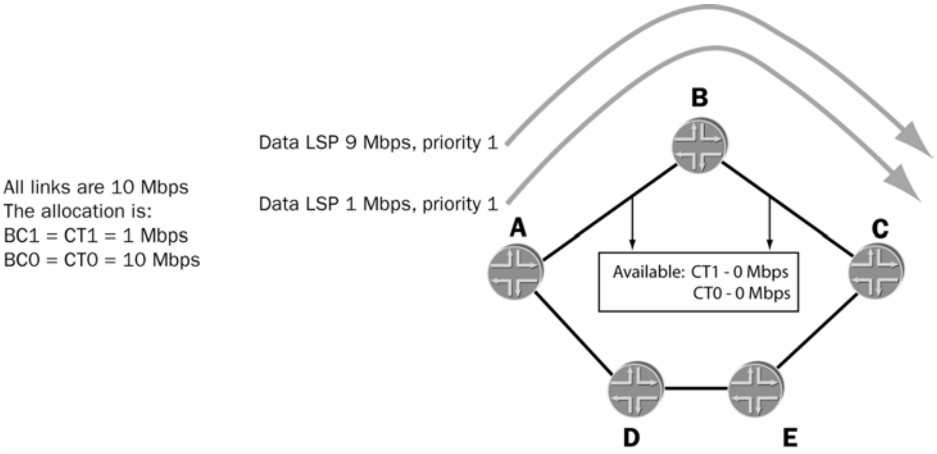


Figure 13a: In the absence of a voice LSP, the second data LSP can establish along the shortest path and use the resources allocated to the voice traffic.

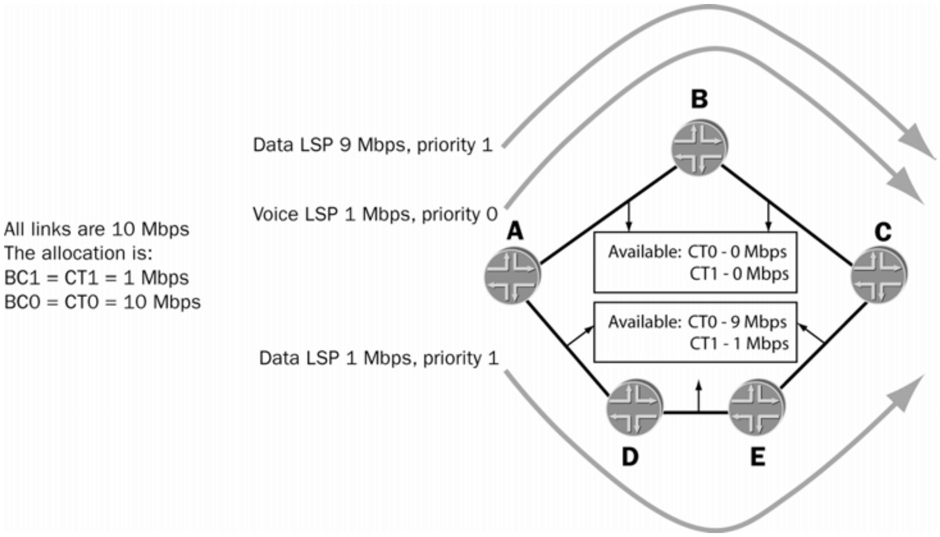


Figure 13b: When using the RDM model, priorities are necessary to guarantee bandwidth to different CTs. When the voice LSP is set up, it will establish on the shortest path and preempt the date LSP.

The calculation of available bandwidth for the RDM model is a bit more complicated, because it must take into account LSPs at several priority levels and from all the CTs that share the particular BC. For example, the available bandwidth for an LSP from CT0 at priority *p* is equal to BC0 minus the allocations for all LSPs from all CTs at priorities better or equal to *p*.

The table below summarizes the differences between MAM and RDM.

MAM	RDM
Maps one BC to one CT; easy to understand and manage	Maps one BC to one or more CTs, less intuitive
Achieves isolation between CTs and guaranteed bandwidth to CTs without the need for preemption	No isolation between CTs; requires preemption to guarantee bandwidth to CTs other than the premium
Bandwidth may be wasted	Efficient use of bandwidth
Useful in networks where preemption is precluded	Not recommended in networks where preemption is precluded

It is clear that the BC model plays a crucial role in determining the bandwidth that is available for each one of the TE classes on a link. The BC model and the bandwidth allocation for each BC are advertised by the IGPs in the Bandwidth Constraints sub-TLV. The IETF does not mandate usage of the same BC model on all links in the network. However, it is easier to configure, maintain, and operate a network where the same bandwidth constraint model is used, and the JUNOS implementation requires consistent configuration of the bandwidth model on all links.

To summarize, the BC model determines the available bandwidth for each CT at each priority level. MAM and RDM are two possible BC models. They differ in the degree of sharing between the different CTs and the degree of reliance on preemption priorities necessary to achieve bandwidth guarantees for a particular CT. The IGPs advertise the BC model and the unreserved bandwidth for the (CT, priority) combinations corresponding to valid TE classes.

4.5 The DiffServ in DiffServ-TE

In the previous sections, we have seen how network resources are partitioned among different types of traffic and how paths with per-traffic-type resource reservations are set up. In the solution presented, the traffic type equates to a desired scheduling behavior, and the available resources for a traffic type are the available resources for a particular scheduler queue. The assumption is that traffic automatically receives the correct scheduling behavior at each hop. This is achieved through DiffServ.

Recall from Section 3.1 that the DiffServ CoS determines the packet's PHB and in particular the scheduling behavior at each hop. In practice, there are two ways to ensure that traffic mapped to a particular DiffServ-TE LSP maps to the correct scheduler queue. The first way is to set the EXP bits appropriately at the LSP ingress (E-LSPs). Recall from Section 3.2 that using E-LSPs, one can support at most eight PHBs, so this solution is good for networks in which less than eight PHBs are required. The second way is to encode the scheduling behavior in the forwarding state (label) installed for the LSP and use the EXP bits to convey the drop preference for the traffic (L-LSP). The scheduling behavior associated with a forwarding entry is signaled at LSP setup time. Any number of PHBs can be supported this way. A combination of both E-LSPs and L-LSPs can be used in a network, assuming that they can be identified (for example, through configuration).

DiffServ provides the correct scheduling behavior for each type of traffic. The combination of DiffServ and per-CT traffic engineering ensures strict service guarantees.

4.5 Tools for Keeping Traffic within Its Reservation Limits

The carefully crafted solution presented in the previous sections would all go to waste if more traffic were forwarded through the LSP than the resources that were allocated for it. In such an event, congestion would occur, queues would be overrun and traffic dropped, with disastrous QoS consequences, not just on the misbehaving LSP, but on all other LSPs from the same CT, crossing the congested link.

The JUNOS implementation provides LSP policers to prevent such a scenario. Policers operate at per-CT granularity at the LSP head-end and ensure that traffic forwarded through an LSP stays within the LSP's bounds. Out-of-profile traffic can be either dropped or marked, affecting the QoS of the misbehaving LSP, but shielding well-behaved LSPs that cross the same links from QoS degradation, as shown in Figure 14. LSP policers make it easy to identify the traffic that needs to be policed, regardless of where traffic is coming from (for example, different incoming interfaces) or going to (for example, different destinations). If the traffic is mapped to the LSP, it will be policed.

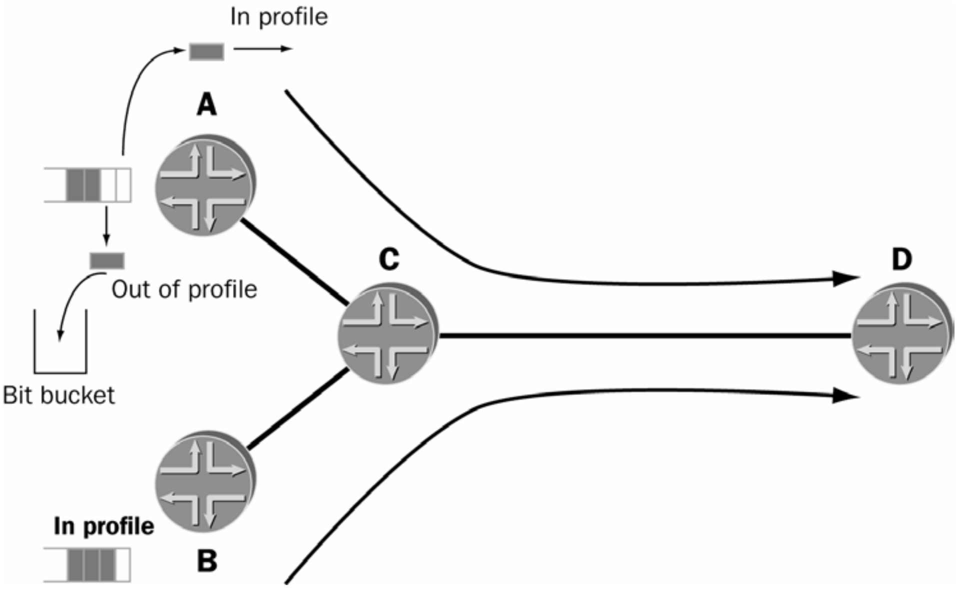


Figure 14: Misbehaving source A has its traffic policed and experiences QoS degradation. The well-behaved traffic from B is not affected.

LSP policing provides a tool for policing traffic that is forwarded through an LSP. But how can one prevent mapping more traffic to an LSP than the LSP can carry?
JUNOS software provides this capability for Martini Layer 2 circuits, in the form of admission control for the Martini Layer 2 circuits. A circuit will not establish unless the underlying RSVP LSP has enough available resources, thus avoiding oversubscription.
For example, if a new Martini Layer 2 circuit requires 20 Mbps bandwidth, the ingress router identifies an LSP that goes to the required destination that has sufficient bandwidth and decrements 20 Mbps from the bandwidth available for other potential Martini Layer 2 circuits that may need to use the LSP in the future.

LSP policing is a simple tool that ensures that traffic stays within the bounds requested for an LSP. Admission control for Martini Layer 2 circuits prevents establishment of Martini Layer 2 circuits over LSPs that don't have enough resources. By keeping traffic within its reservation limits, QoS can be enforced.

4.6 Deploying the DiffServ-TE Solution

To summarize the previous sections, the following steps are required to deploy a DiffServ-TE solution:

1) Decide on a BC model and the bandwidth associated with each BC on each link.
2) Configure the buffer and bandwidth allocations on each link to be consistent with

Step 1.
3) Decide which CTs and priorities are required.
4) Choose an IGP.
5) Configure LSPs with the desired bandwidth reservation, CT, and priority.
6) Configure policers and bandwidth booking, if required.
7) Decide whether the DiffServ treatment will be determined from the EXP bits or the label. If the DiffServ treatment is based on the EXP bits, configure the EXP-to-PHB mappings consistently throughout the DiffServ domain and make sure the traffic is marked correctly.

Let us briefly look at the migration of a traffic engineered network to DiffServ-TE. As a first step, the network operator must decide which combinations of CTs and priorities are required in the network. Recall from Section 4.1 that LSPs with no per-CT requirements are mapped to CT0. Therefore, in a migration scenario, the combinations of CT0 and of the priorities used for TE LSPs that already exist in the network must be selected as valid combinations. The second step is to map the (CT, priority) combinations selected in the first step to TE classes. Recall from Section 4.2 that the Unreserved Bandwidth TLV is overwritten with the per-TE-class information. Network migrations are typically done in stages, so there will be both old and new nodes advertising the Unreserved Bandwidth TLV to each other, but with different semantics. Old nodes will fill in field *i* of the Unreserved Bandwidth TLV the available bandwidth for (CT0, *i*). New nodes will fill the available bandwidth for TE*i*. To provide a consistent picture of the available resources to both old and new nodes, (CT0, *i*) must map to TE*i*. Such a definition ensures smooth interoperation between nodes that support the DiffServ-TE extensions and nodes that do not.

5. Advantages of the JUNOS Implementation

JUNOS software supports the DiffServ-TE standards defined in the IETF. Several advantages of the JUNOS implementation are as follows:

- A completely interoperable, standards-based solution that can operate in a mixed environment and does not require an upgrade of the entire network.
- High availability features, making the platforms suited to carrying mission-critical traffic.
- A strong DiffServ implementation
    - The code base is mature and proven in actual deployments (first released in JUNOS 5.1).
    - RED profiles are independent of the interface speed.
    - Latency and jitter for queues that do not exceed their bandwidth allocation is very low, even under conditions of extreme congestion.

  - DiffServ Code Point aliases make configuration easy.
- Very robust routing and signaling protocol implementation.
- Proven scalability of the MPLS-TE implementation, providing a solid foundation for DiffServ-TE.
- Resiliency in case of failure. The path protection, fast-reroute, and link-node protection mechanisms available in the JUNOS software ensure resiliency in case of failure.
- LSP policing offers a simple mechanism to police traffic and prevent a misbehaving source from degrading the QoS guarantees for other traffic except its own.
- Admission control for Martini Layer 2 circuits ensures that circuits are only established when bandwidth can be guaranteed.
- Support for several bandwidth models gives network operators the flexibility to pick the one that best suits their needs.

Implementation of the DiffServ-TE technology requires extensions to the IGPs, RSVP, and the CSPF algorithm. These extensions are backwards compatible and allow staged deployment of the technology in the network.

6. Conclusion

The DiffServ-TE technology defined by the IETF provides strict service guarantees for different traffic types. Such guarantees allow service providers to deploy new services and achieve better resource utilization for existing services. The JUNOS implementation offers a standards-based, interoperable, and scalable implementation of DiffServ-TE, along with unique tools for ensuring that traffic stays within the limits of the resources that were reserved for it.

7. References

[DSTE- MAM] - Le Faucheur F., Lai K., *Maximum Allocation Bandwidth Constraints Model for DS-TE* - draft-ietf-tewg-diff-te-mam-01.txt

[DSTE-PROTO] - Le Faucheur F. et al, *Protocol extensions for support of DS-aware MPLS TE* - draft-ietf-tewg-diff-te-proto-05.txt

[DSTE-RDM] - Le Faucheur F. et al, *Russian Dolls Bandwidth Constraints Model for DS-TE* - draft-ietf-tewg-diff-te-russian-04.txt

[FRR] - Pan P. et al, *Fast Reroute Extensions to RSVP-TE for LSP Tunnels* - draft-ietf-mpls-rsvp-lsp-fastreroute-03.txt, work in progress

[ISIS-TE] – Smit H., Li T., *IS-IS extensions for Traffic Engineering* - draft-ietf-isis-traffic-05.txt

[MPLSTECH] – Davie B., Rekhter Y. *, MPLS Technology and Applications*

[OSPF-TE] – Katz D. Yeung D., *Traffic Engineering Extensions to OSPF* - draft-katz-yeung-ospf-traffic-10.txt

[RFC2475] – Blake S. et al, *An Architecture for Differentiated Services* – RFC 2475

[RFC2702] - Awduche D. et al, *Requirements for Traffic Engineering Over MPLS* – RFC 2702

[RFC3031] - Rosen E. et al, *Multiprotocol Label Switching Architecture* – RFC 3031

[RFC3036] – Anderson L. et al, *LDP Specification* – RFC 3036

[RFC3209] - Awduche D. et al, *RSVP-TE: Extensions to RSVP for LSP Tunnels* - RFC 3209

[RFC3270] - Le Faucheur F. et al, *MPLS Support of Diff-Serv* - RFC 3270

[RFC3468] – Anderson L., Swallow G., *The Multiprotocol Label Switching (MPLS) Working Group decision on MPLS signaling protocols* – RFC 3468

[RFC3564]- Le Faucheur F. et al, *Requirements for Support of Differentiated Services-aware MPLS Traffic Engineering* - RFC 3564

Acknowledgements

The author would like to thank Yakov Rekhter, Julian Lucek and Amir Tabdili for their comments, suggestions and insight; Mike Capuano, Hannes Gredler, Nischal Sheth and Chelian Pandian for their review; Aviva Garret for editing; and Susan Lane for production support.

Juniper Networks assumes no responsibility for any inaccuracies in this document. Juniper Networks reserves the right to change, modify, transfer, or otherwise revise this publication without notice.