INTERNET-DRAFT
Expires: February 2004

Roland Bless
Univ. of Karlsruhe
Klaus Wehrle
Univ. of Karlsruhe/ICSI

Internet Draft

August 2003

Document: draft-bless-diffserv-multicast-07.txt

IP Multicast in Differentiated Services Networks
<draft-bless-diffserv-multicast-07.txt>

Status of this Memo

Abstract

This document discusses the problems of IP Multicast use in Differentiated Services (DS) networks, expanding on the discussion in RFC 2475 ("An Architecture of Differentiated Services"). It also suggests possible solutions to these problems, describes a potential implementation model, and presents simulation results.

Table of Contents

1 Introduction

This document discusses the problems of IP Multicast use in Differentiated Services (DS) networks, expanding on the discussion in RFC 2475 ("An Architecture of Differentiated Services"). It also suggests possible solutions to these problems, describes a potential implementation model, and presents simulation results.

The "Differentiated Services" (DiffServ or DS) approach [1, 2, 3] defines certain building blocks and mechanisms to offer qualitatively better services than the traditional best-effort delivery service in an IP network. In the DiffServ Architecture [2] scalability is achieved by avoiding complexity and maintenance of per-flow state information in core nodes and by pushing unavoidable complexity to the network edges. Therefore, individual flows belonging to the same service are aggregated, thereby eliminating the need for complex classification or managing state information per flow in interior nodes.

On the other hand, the reduced complexity in DS nodes makes it more complex to use those "better" services together with IP Multicast (i.e., point-to-multipoint or multipoint-to-multipoint communication). Problems emerging from this fact are described in section 2. Although the basic DS forwarding mechanisms also work with IP Multicast, some facts have to be considered which are related to the provisioning of multicast resources. However, it is important to integrate IP Multicast functionality right from the beginning into the architecture, and, to provide simple solutions for those problems not defeating the gained advantages so far.

1.1 Management of Differentiated Services

At least for Per-Domain Behaviors and services based on the EF PHB, admission control and resource reservation are required. Furthermore, installation and updating of traffic profiles in boundary nodes is necessary. Most network administrators cannot accomplish this task manually, even for long term service level agreements (SLAs). Furthermore, offering services on demand requires some kind of signaling and automatic admission control procedures.

However, no standardized resource management architecture for DiffServ domains exists. So for the rest of the document, it is assumed that at least some logical resource management entity is available that performs resource-based admission control and allotment functions. This entity may also be realized in a distributed fashion, e.g., within the routers themselves. Detailed

aspects of the resource management realization within a DiffServ domain as well as the interactions between resource management and routers or end-systems (e.g., signaling for resources) are out of scope of this document.

Protocols for signaling a reservation request to a Differentiated Services Domain are required. For accomplishing end-system signaling to DS domains RSVP [4] may be used with new DS specific reservation objects [5]. RSVP provides support for multicast scenarios and is already supported by many systems. However, application of RSVP in a DiffServ multicast context may lead to problems that are also described in the next section.

2 Problems of IP Multicast in DS Domains

Although potential problems and the complexity of providing multicast with Differentiated Services are considered in a separate section of [2], both aspects have to be discussed in greater detail. The simplicity of the DiffServ Architecture and its DS node types is necessary to reach high scalability, but it causes also fundamental problems in conjunction with the use of IP Multicast in DS domains. The following subsections describe these problems for which a generic solution is proposed in section 3. This solution is as scalable as IP Multicast and needs no resource separation by using different codepoint values for unicast and multicast traffic.

Because Differentiated Services are unidirectional by definition, the point-to-multipoint communication is also considered as unidirectional. In traditional IP Multicast any node can send packets spontaneously and asynchronously to a multicast group specified by their multicast group address. I.e., traditional IP Multicast offers a multipoint-to-multipoint service, also referred to as Any-Source Multicast. Implications of this feature are discussed in section 2.3.

For subsequent considerations we assume, unless stated otherwise, at least a unidirectional point-to-multipoint communication scenario in which the sender generates packets which experience a "better" Per-Hop Behavior than the traditional default PHB, resulting in a service of better quality than the default best-effort service. In order to accomplish this, a traffic profile corresponding to the traffic conditioning specification has to be installed in the sender's first DS-capable boundary node. Furthermore, it must be assured that the corresponding resources are available on the path from the sender to all the receivers, possibly requiring adaptation of traffic profiles at involved domain boundaries. Moreover, on demand resource reservations may be receiver-initiated, too.

2.1 Neglected Reservation Subtree Problem (NRS Problem)

Typically, resources for Differentiated Services must be reserved before actually using them. But in a multicast scenario group membership is often highly dynamic, therefore limiting the use of a sender-initiated resource reservation in advance. Unfortunately, dynamic addition of new members of the multicast group using Differentiated Services can adversely affect other existing traffic, if resources were not explicitly reserved before use. A practical proof of this problem is given in section 8.

IP Multicast packet replication usually takes place when the packet is handled by the forwarding core (cf. Fig. 1), i.e., when it is forwarded and replicated according to the multicast forwarding table. Thus, a DiffServ capable node would also copy the content of the DS field [1] into the IP packet header of every replicate. Consequently, replicated packets get exactly the same DS codepoint (DSCP) as the original packet, and, therefore experience the same forwarding treatment as the incoming packets of this multicast group. This is also illustrated in Fig. 1 where each egress interface comprises functions for (BA-) classification, traffic conditioning, and queueing.

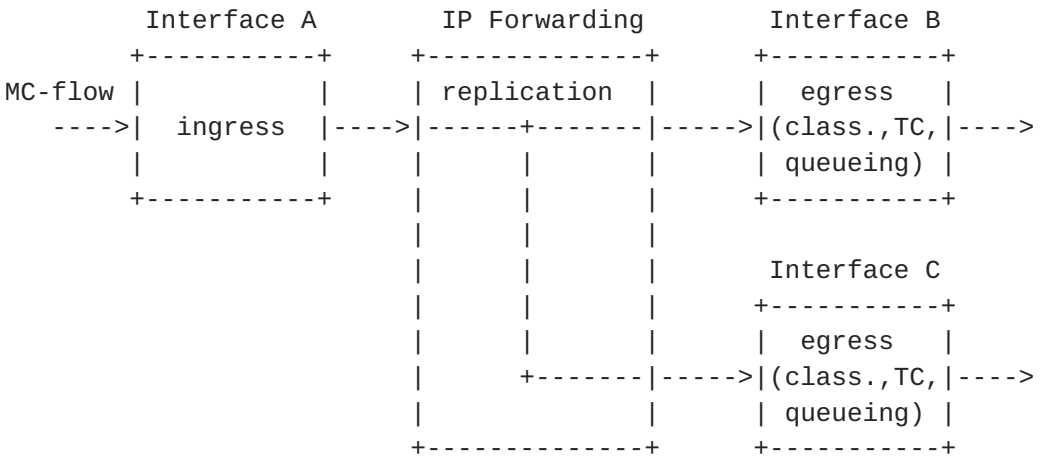


Figure 1: Multicast packet replication in a DS node

Normally, the replicating node cannot test whether a corresponding resource reservation exists for a particular flow of replicated packets on an output link (i.e., its corresponding interface). This is caused by the fact that flow-specific information (e.g., traffic profiles) is usually not available in every boundary and interior node.

When a new receiver joins an IP Multicast group, a multicast routing

protocol (e.g., DVMRP [6], PIM-DM [7] or PIM-SM [8]) grafts a new

branch to an existing multicast tree in order to connect the new receiver to the tree. As a result of tree expansion and missing per-flow classification and policing mechanisms, the new receiver will implicitly use the service of better quality, because of the copied "better" DSCP.

If the additional amount of resources which are consumed by the new part of the multicast tree are not taken into account by the domain resource management (cf. section 1.1), the currently provided level of quality of service of other receivers (with correct reservations) will be affected adversely or even violated. This negative effect on existing traffic contracts by a neglected resource reservation -- in the following designated as Neglected Reservation Subtree Problem (NRS Problem) -- must be avoided under all circumstances.

One can distinguish two distinct major cases of the NRS Problem. They show a different behavior depending on the location of the branching point. In order to compare their different effects a simplistic example of a share of bandwidth is illustrated in Fig. 2 and is used in the following explanations. Neither the specific PHB types nor their assigned bandwidth share are important, whereas their relative priority with respect to each other is of importance.

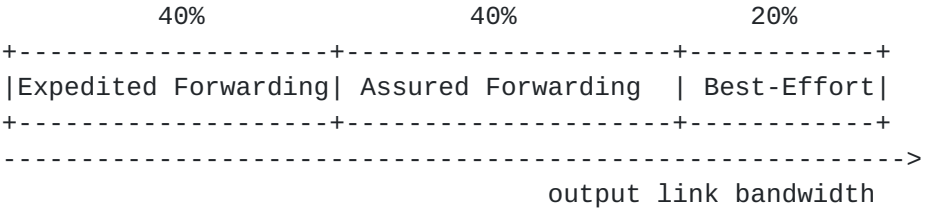


Figure 2: An example bandwidth share of different behavior aggregates

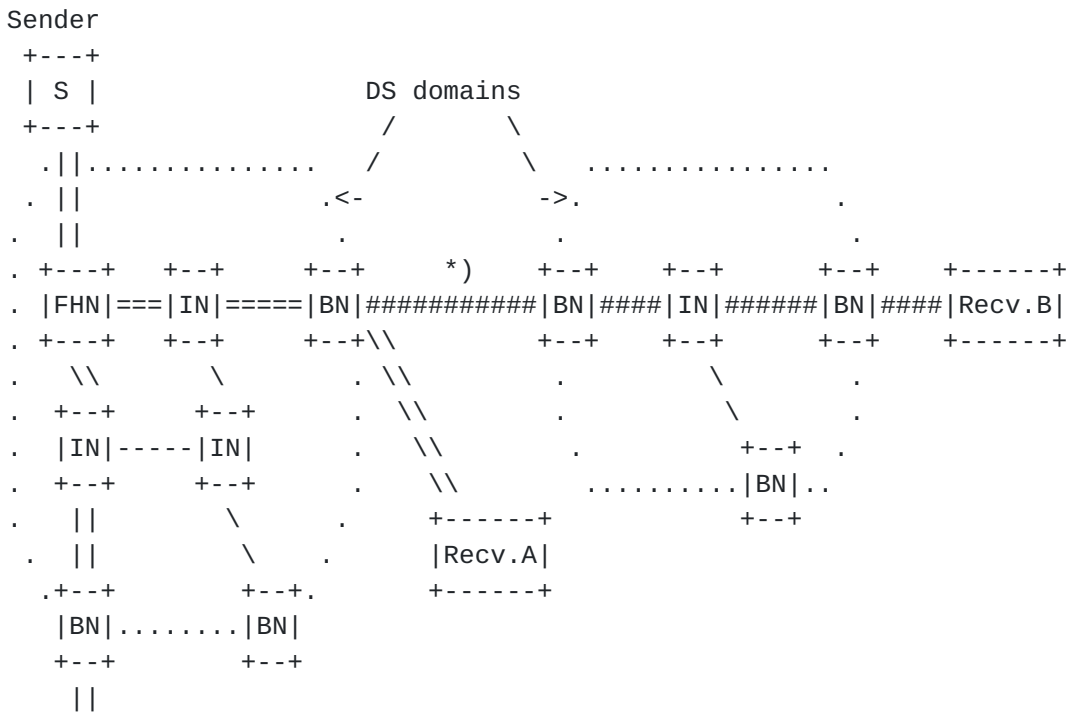
The bandwidth of the considered output link is shared by three types of services (i.e., by three behavior aggregates): Expedited Forwarding, Assured Forwarding and the traditional Best-Effort service. In this example we assume that routers perform simple priority queueing, where EF has the highest, AF a middle, and Best-Effort the lowest assigned priority. Were Weighted Fair Queueing (WFQ) to be used, the described effects would essentially also occur, only with minor differences. In the following scenarios it is illustrated that PHBs of equal or lower priority (in comparison to the multicast flow's PHB) are affected by the NRS problem.

The Neglected Reservation Subtree problem appears in two different cases:

o Case 1: If the branching point of the new subtree (at first only a

branch) and the previous multicast tree is an (egress) boundary

node, as shown in Fig. 3, the additional multicast flow now increases the total amount of used resources for the corresponding behavior aggregate on the affected output link. The total amount will be greater than the originally reserved amount. Consequently, the policing component in the egress boundary node drops packets until the traffic aggregate is in accordance to the traffic contract. But during dropping packets, the router can not identify the responsible flow (because of missing flow classification functionality), and, thus randomly discards packets, whether they belong to a correctly behaving flow or not. As a result, there will be no longer any service guarantee for the flows with properly reserved resources.



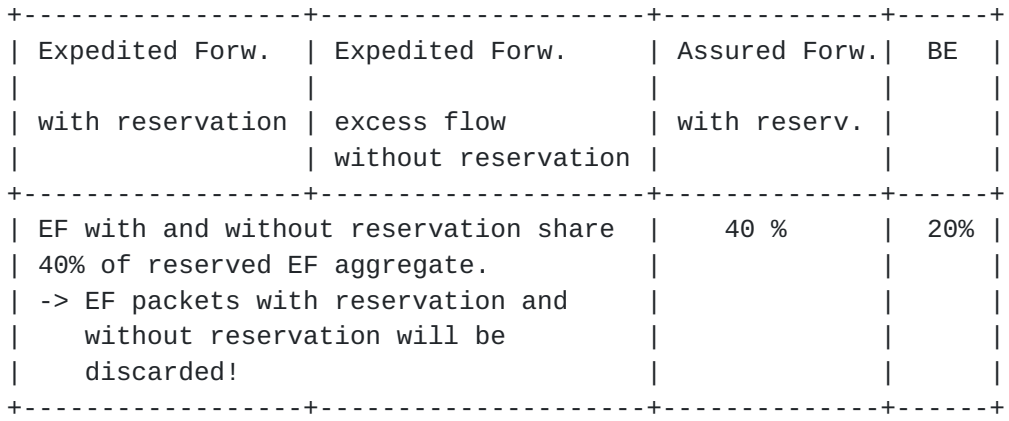
S: Sender
Recv.x: Receiver x
FHN: First-Hop Node
BN: Boundary Node
IN: Interior Node
===: Multicast branch with reserved bandwidth
###: Multicast branch without reservation
*) Bandwidth of EF aggregated on the output link is higher than actual reservation, EF aggregate will be limited in bandwidth without considering the responsible flow.

Figure 3: The NRS Problem (case 1) occurs when Receiver B joins

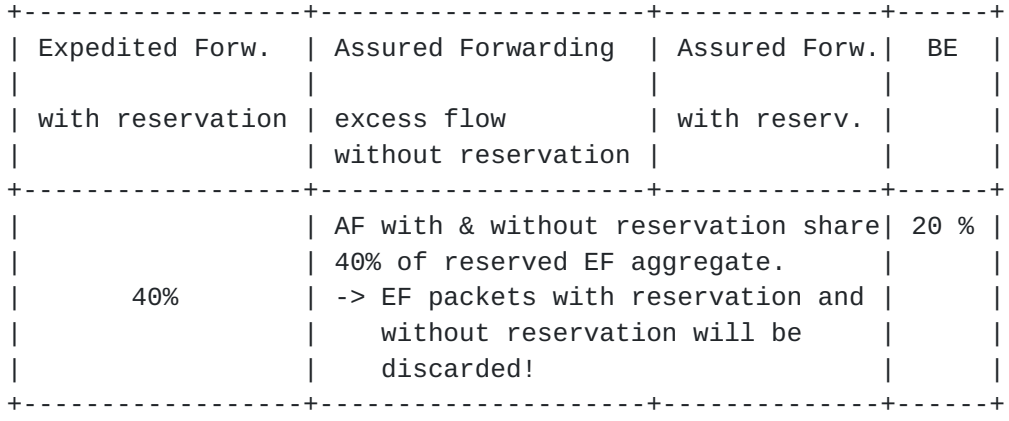
Fig. 3 describes this situation: it is assumed that receiver A is

already attached to the egress boundary node (BN) of the first

domain. Furthermore, resources are properly reserved along the path to receiver A and packets that are marked correspondingly. When receiver B joins the same group as receiver A, packets are replicated and forwarded along the new branch towards the second domain with the same PHB as for receiver A. If this PHB is EF, the new branch possibly exhausts allotted resources for the EF PHB, adversely affecting other EF users that receive their packets over the link that is marked with the *). The BN usually ensures that outgoing traffic aggregates to the next domain are conforming to the agreed traffic conditioning specification. The egress BN will, therefore, drop packets of the PHB type that is used for the multicast flow. Other PHBs of lower or higher priority are not affected adversely in this case. The following example in Fig. 4. illustrates this for two PHBs.



(a) Excess flow has EF codepoint



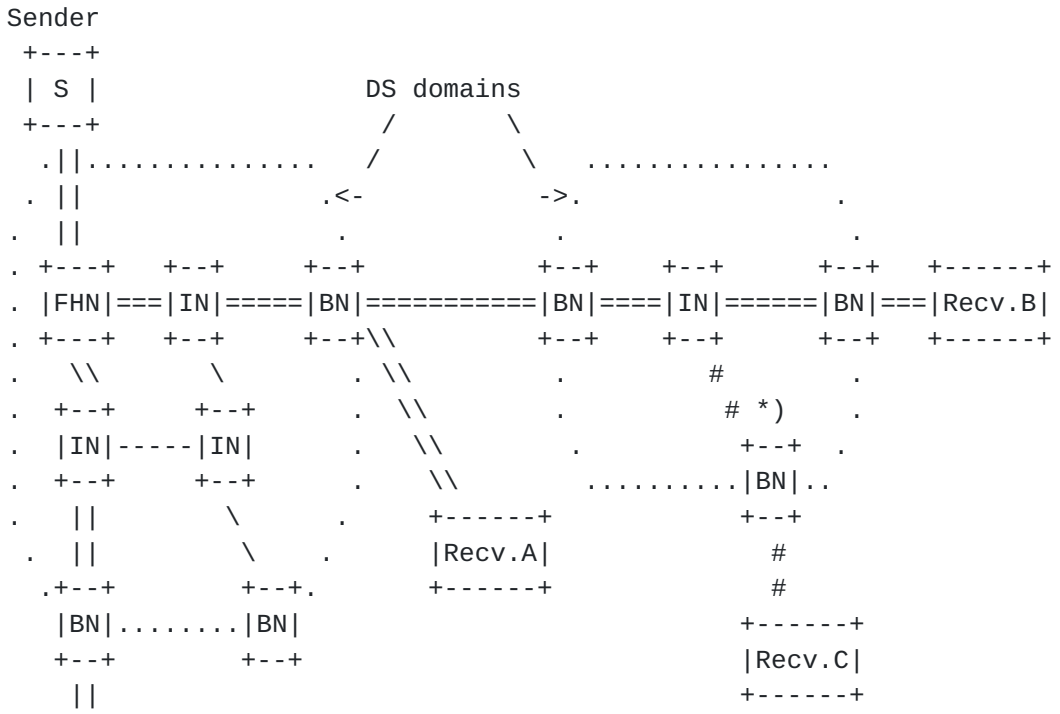
(b) Excess flow has AF codepoint

Figure 4: Resulting share of bandwidth in a egress boundary node with a neglected reservation of (a) an Expedited Forwarding flow or (b) an Assured Forwarding flow.

Fig. 4 shows the resulting share of bandwidth in cases when (a) Expedited Forwarding and (b) Assured Forwarding is used by the additional multicast branch causing the NRS Problem. Assuming that the additional traffic would use another 30% of the link bandwidth, Fig. 4 (a) illustrates that the resulting aggregate of Expedited Forwarding (70% of the outgoing link bandwidth) is throttled down to its originally reserved 40%. In this case, the amount of dropped EF bandwidth is equal to the amount of excess bandwidth. Consequently the original Expedited Forwarding aggregate (which had 40% of the link bandwidth reserved) is affected by packet losses, too. The other services, e.g., Assured Forwarding or Best-Effort, are not disadvantaged.

Fig. 4 (b) shows the same situation for Assured Forwarding. The only difference is that now Assured Forwarding is solely affected by discards, the other services will still get their guarantees. In either case, packet losses are restricted to the misbehaving service class by the traffic meter and policing mechanisms in boundary nodes. Moreover, the latter problem (case 1) occurs only in egress boundary nodes, because they are responsible, that not more traffic is leaving the Differentiated Services domain, than the following ingress boundary node will accept. Therefore, those violations of SLAs will be already detected and processed in egress boundary nodes.

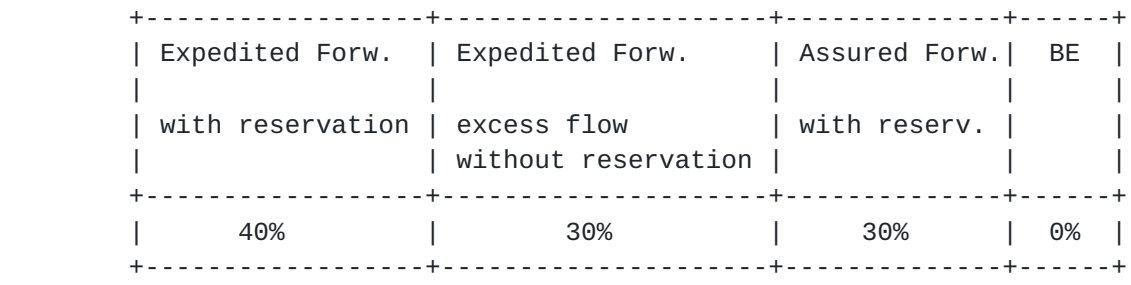
- Case 2: The Neglected Reservation Subtree problem can also occur, if the branching point between the previous multicast tree and the new subtree is located in an interior node (as shown in Fig. 5). In Fig. 5 it is assumed that receivers A and B have already joined the multicast group and have reserved resources accordingly. The interior node in the second domain starts replication of multicast packets as soon as receiver C joins. Because the router is not equipped with metering or policing functions it will not recognize any amount of excess traffic and will forward the new multicast flow. If the latter belongs to a higher priority service, such as Expedited Forwarding, bandwidth of the aggregate is higher than the aggregate's reservation at the new branch and will use bandwidth from lower priority services.



FHN: First-Hop Node, BN: Boundary Node, Recv.x: Receiver x
S: Sender, IN: Interior Node
===: Multicast branch with reserved bandwidth
###: Multicast branch without reservation
*) Bandwidth of EF aggregated on the output link is higher than actual reservation, EF aggregate will be limited in bandwidth without considering the responsible flow

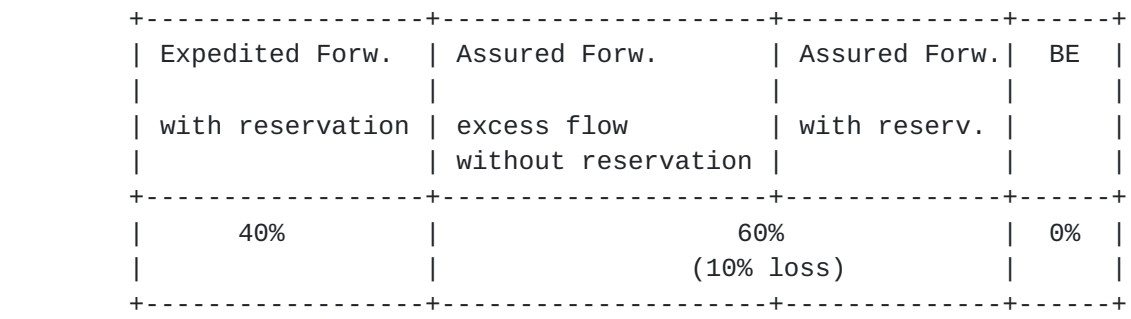
Figure 5: Neglected Reservation Subtree problem case 2 after join of receiver C

The additional amount of EF without a corresponding reservation is forwarded together with the aggregate which has a reservation. This results in no packets losses for Expedited Forwarding as long as the resulting aggregate is not higher than the output link bandwidth. Because of its higher priority, Expedited Forwarding gets as much bandwidth as needed and as is available. The effects on other PHBs are illustrated by the following example in Fig. 6.



EF with reservation and the excess flow use together 70% of the link bandwidth, because EF (with or without reservation has the highest priority.

(a) Excess flow has EF codepoint



AF with reservation and the excess flow use together 60% of the link bandwidth, because EF has the highest priority (-> 40%). 10% of AF packets will be lost.

(b) Excess flow has AF codepoint

Figure 6: Resulting share of bandwidth in an interior node with a neglected reservation of (a) a Expedited Forwarding flow or (b) an Assured Forwarding flow

The result of case 2 is, that there is no restriction for Expedited Forwarding, but as Fig. 6 (a) shows, other services will be extremely disadvantaged by this use of non-reserved resources. Their bandwidth is used by the new additional flow. In this case, the additional 30% Expedited Forwarding traffic preempts resources from the Assured Forwarding traffic, which in turn preempts resources from the best-effort traffic, resulting in 10% packet losses for the Assured Forwarding aggregate and complete loss of best-effort traffic. The example in Fig. 6 (b) shows that this can also happen with lower priority services like Assured Forwarding. When a reservation for a service flow with lower priority is neglected, other services (with even lower priority) can be reduced in their quality (in this case the best-effort service). As shown in the example, the service's aggregate causing the NRS

problem can itself be affected by packet losses (10% of the

Assured Forwarding aggregate is discarded). Besides the described problems of case 2, case 1 will occur in the DS boundary node of the next DS domain, that performs traffic metering and policing for the service aggregate.

Directly applying RSVP to Differentiated Services would also result in an temporary occurrence of the NRS Problem. A receiver has to join the IP multicast group to receive the sender's PATH messages, before being able to send a resource reservation request (RESV message). Thus, the join for receiving PATH messages can cause the NRS Problem, if this situation is not handled in a special way (e.g., by marking all PATH messages with codepoint 0 and filtering or re-marking all other data packets of the multicast flow).

2.2 Heterogeneous Multicast Groups

Heterogeneous multicast groups contain one or more receivers, which would like to get another service or quality of service as the sender provides or other receiver subsets currently use. A very important characteristic which should be supported by Differentiated Services is that participants requesting a best-effort quality only should also be able to participate in a group communication which otherwise utilizes a better service class. The next better support for heterogeneity provides concurrent use of more than two different service classes within a group. Things tend to get even more complex when not only different service classes are required, but also different values for quality parameters within a certain service class.

A further problem is to support heterogeneous groups with different service classes in a consistent way. It is possible that some services will not be comparable to each other so that one service cannot be replaced by the other and both services have to be provided over the same link within this group.

Because an arbitrary new receiver that wants to get the different service can be grafted to any point of the current multicast delivery tree, even interior nodes may have to replicate packets using the different service. At a first glance, this seems to be a contradiction with respect to simplicity of the interior nodes, because they do not even have any profile available and should now convert the service quality of individual receivers. Consequently, in order to accomplish this, interior nodes have to change the codepoint value during packet replication.

2.3 Dynamics of Any-Source Multicast

Basically, within an IP multicast group any participant (actually, it can be any host not even receiving packets of this multicast group) can act as a sender. This is an important feature which should also be available in case a specific service other than best-effort is used within the group. Differentiated Services possess conceptually a unidirectional character. Therefore, for every multicast tree implied by a sender, resources must be reserved separately if simultaneous sending should be possible with a better service. This is even true if shared multicast delivery trees are used (e.g., with PIM-SM or Core Based Trees). If not enough resources are reserved for a service within a multicast tree allowing simultaneous sending of more than one participant, the NRS problem will occur again. The same argument applies to half-duplex reservations which would share the reserved resources by several senders, because it cannot be ensured by the network that exactly one sender sends packets to the group. Accordingly, the corresponding RSVP reservation styles "Wildcard Filter" and "Shared-Explicit Filter" [4] cannot be supported within Differentiated Services. The Integrated Services approach is able to ensure the half-duplex nature of the traffic, because every router can check each packet for its conformance with the installed reservation state.

3 Solutions for Enabling IP-Multicast in Differentiated Services Networks

The problems described in the previous section are mainly caused by the simplicity of the Differentiated Services architecture. Solutions have to be developed which do not introduce additional complexity which would otherwise diminish the scalability of the DiffServ approach. This document suggests a straightforward solution for most of the problems.

3.1 Solution for the NRS Problem

The proposed solution consists conceptually of the following three steps that are described in more detail later.

1. A new receiver joins a multicast group that is using a DiffServ service. Multicast routing protocols accomplish the connection of the new branch to the (possibly already existing) multicast delivery tree as usual.

2. The unauthorized use of resources is avoided by re-marking at branching nodes all additional packets leaving down the new branch. At first, the new receiver will get all packets of the multicast group without quality of service. The management

entity of the correspondent DiffServ domain may get informed about the extension of the multicast tree.

3. If a pre-issued reservation is available for the new branch or somebody (receiver, sender or a third party) issues one, the management entity instructs the branching router to set the corresponding codepoint for the demanded service.

Usage of resources which were not reserved before must be prevented. In the following discussed example, the case is considered when the join of a new receiver to a DS multicast group requires grafting of a new branch to an already existing multicast delivering tree. The connecting node that joins both trees converts the codepoint (and therefore the Per-Hop Behavior) to a codepoint of a PHB which is similar to the default PHB in order to provide a best-effort-like service for the new branch. More specifically, this particular PHB can provide a service that is even worse than the best-effort service of the default PHB.

The conversion to this specific PHB could be necessary in order to avoid unfairness being introduced otherwise within the best-effort service aggregate, and, which results from the higher amount of resource usage of the incoming traffic belonging to the multicast group. If the rate at which re-marked packets are injected into the outgoing aggregate is not reduced, those re-marked packets will probably cause discarding of other flow's packets in the outgoing aggregate if resources are scarce.

Therefore, the re-marked packets from this multicast group should be discarded more aggressively than other packets in this outgoing aggregate. This could be accomplished by using an appropriate configured PHB (and a related DSCP) for those packets. In order to distinguish this kind of PHB from the default PHB, it is referred to as Limited Effort (LE) PHB (which can be realized by an appropriately configured AF PHB [9] or Class Selector Compliant PHB [1]) throughout this document. Merely dropping packets more aggressively at the re-marking node is not sufficient, because there may be enough resources in the outgoing behavior aggregate (BA) to transmit every re-marked packet and not requiring discarding any other packets within the same BA. However, resources in the next node may be short for this particular BA. Those "excess" packets, therefore, must be identifiable at this node.

Re-marking packets is only required at branching nodes, whereas all other nodes of the multicast tree (such with outdegree 1) replicate packets as usual. Because a branching node may also be an interior node of a domain, re-marking of packets requires conceptually per-flow classification. Though this seems to be in contradiction to the DiffServ philosophy of a core that avoids per-flow states, IP

multicast flows are different from unicast flows: traditional IP

multicast forwarding and multicast routing require to install states per multicast group for every outgoing link anyway. Therefore, re-marking in interior nodes is to the same extent scalable as IP multicast is (cf. section 4).

Re-marking with standard DiffServ mechanisms [10] for every new branch requires activation of a default traffic profile. The latter accomplishes re-marking by using a combination of an MF-classifier and a marker at an outgoing link that constitutes a new branch. The classifier will direct all replicated packets to a marker that sets the new codepoint. An alternative implementation is described in section 7.

The better service will only be provided if a reservation request was processed and approved by the resource management function. That means an admission control test must be performed before resources are actually used by the new branch. In case the admission test is successful, the re-marking node will be instructed by the resource management to stop re-marking and to use the original codepoint again (conceptually by removing the profile).

In summary, only those receivers will obtain a better service within a DiffServ multicast group, which previously reserved the corresponding resources in the new branch with assistance of the resource management. Otherwise they get a quality which might be even lower than best-effort.

3.2 Solution for Supporting Heterogeneous Multicast Groups

In this document considerations are limited to provisioning different service classes, but not different quality parameters within a certain service class.

The proposed concept from section 3.1 provides also a limited solution of the heterogeneity problem. Receivers are allowed to obtain a Limited Effort service without a reservation, so that at least two different service classes within a multicast group are possible. Therefore, it is possible that any receiver may participate in the multicast session without getting any quality of service. This is useful if a receiver just wants to see whether the content of the multicast group is interestingly enough, before requesting a better service which must be paid for (like snooping into a group without prior reservation).

Alternatively, a receiver might not be able to receive this better quality of service (e.g., because it is mobile and uses a wireless link), but it may be satisfied with the reduced quality, instead of getting no content at all.

Additionally, applying the RSVP concept of listening for PATH messages before sending any RESV message is now feasible again. Without using the proposed solution this would have caused the NRS Problem.

Theoretically, the proposed approach also supports more than two different services within one multicast group, because the additional field in the multicast routing table can store any DSCP value. However, this would work only if PHBs can be ordered, so that the "best" PHB among different required PHBs downstream is chosen to be forwarded on a specific link. This is mainly a management issue and out of scope for this document.

3.3 Solution for Any-Source Multicast

Every participant would have to initiate an explicit reservation if he wants to make sure that it is possible to send with a better service quality to the group, regardless whether other senders within the group already use the same service class simultaneously. This would require a separate reservation for each sender-rooted multicast tree.

However, in the specific case of best-effort service (the default PHB), it is nevertheless possible for participants to send packets anytime to the group without requiring any additional mechanisms. The reason for this is that the first DS-capable boundary node will mark those packets with the DSCP of the default PHB because of a missing traffic profile for this particular sender. The first DS capable boundary nodes should therefore always classify multicast packets based on both the sender's address and the multicast group address.

4 Scalability Considerations

The proposed solution does not add complexity to the DS architecture or to a DS node, and, it does not change the scalability properties of DiffServ. With current IP multicast routing protocols a multicast router has to manage and hold state information per traversing multicast flow. The suggested solution scales to the same extent as IP multicast itself, because the proposed re-marking may occur per branch of a multicast flow. This re-marking is logically associated with an addition to the multicast routing state that is required anyway. In this respect, re-marking of packets for multicast flows in interior nodes is not considered as a scalability problem or to be in contradiction to the DiffServ approach itself. It is important to distinguish the multicast case from existing justifiable scalability concerns relating to re-marking packets of unicast flows within interior routers. Moreover, the decision when to change a re-marking policy is not performed by the router, but by some

management entity at a time scale which is different from the time scale at the packet forwarding level.

5 Deployment Considerations

The solution proposed in section 3.1 and can be deployed on most nowadays available router platforms. Especially architectures that perform routing and forwarding functions in software could be updated by a new software release.

However, there may be some specialized hardware platforms which could currently not be able to deploy the proposed solution from section 7. This may be the case when a multicast packet is directly duplicated on the backplane of the router, so that all outgoing interfaces read the packet in parallel. Consequently, the codepoint cannot be changed for a subset of these outgoing interfaces and the NRS problem can not be solved directly in the branching point.

In this case, there exist several alternative solutions:

1. As mentioned in section 3.1, if traffic conditioning mechanisms can be applied on the outgoing packets at the individual output interfaces, a combination of classifier and marker may be used for each branch.

2. The change of the codepoint for subtrees without properly allocated resources could take place in the following downstream router. There, for every incoming packet of the considered multicast group, the codepoint would be changed to the value that the previous router should have set. If a LAN (e.g., a high-speed switching LAN) is attached to the considered outgoing interface, then on every router connected to the LAN, packets of the considered group should be changed on the incoming interface by standard DiffServ mechanisms.

Future releases of router architectures may support the change of the codepoint directly in the replication process as proposed in section 7.

6 Security Considerations

Basically, the security considerations in [1] apply. The proposed solution does not imply new security aspects. If a join of arbitrary end-systems to a multicast group is not desired (thereby receiving a lower than best-effort quality) the application usually has to exclude these participants. This can be accomplished by using authentication, authorization or ciphering techniques at application level -- like in traditional IP multicast scenarios.

Moreover, it is important to consider the security of corresponding management mechanisms, because they are used to activate re-marking of multicast flows. On the one hand, functions for instructing the router to mark or re-mark packets of multicast flows are attractive targets to perform theft of service attacks. On the other hand, their security depend on the router management mechanisms which are used to realize this functionality. Router management should generally be protected against unauthorized use, therefore preventing those attacks as well.

7 Implementation model example

One possibility to implement the proposed solution from section 3.1 is described in the following. It has to be emphasized that other realizations are also possible, and, this description should not be understood as a restriction on potential implementations. The benefit of the following described implementation is, that it does not require any additional classification of multicast groups within an aggregate. It serves as a proof of concept that no additional complexity is necessary to implement the proposed general solution described in section 3.

Because every multicast flow has to be considered by the multicast routing process (in this context, this notion signifies the multicast forwarding part and not the multicast route calculation and maintenance part, cf. Fig. 1), the addition of an extra byte in each multicast routing table entry containing the DS field, and, thus its DS codepoint value, per output link (resp. virtual interface, see Fig. 8) results in nearly no additional cost. Packets will be replicated by the multicast forwarding process, so this is also the right place for setting the correct DSCP values of the replicated packets. Their DSCP values are not copied from the incoming original packet, but from the additional DS field in the multicasting routing table entry for the corresponding output link (only the DSCP value must be copied, while the two remaining bits are ignored and are present for simplification reasons only). This field contains initially the codepoint of the LE PHB if incoming packets for this specific group do not carry the codepoint of the default PHB.

When a packet arrives with the default PHB, the outgoing replicates should also get the same codepoint in order to retain the behavior of nowadays common multicast groups using the default PHB. A router configuration message changes the DSCP values in the multicast routing table and may also carry the new DSCP value which should be set in the replicated packets. It should be noted that although re-marking may also be performed by interior nodes, the forwarding performance will not be decreased, because the decision when and what to re-mark is made by the management (control plane).

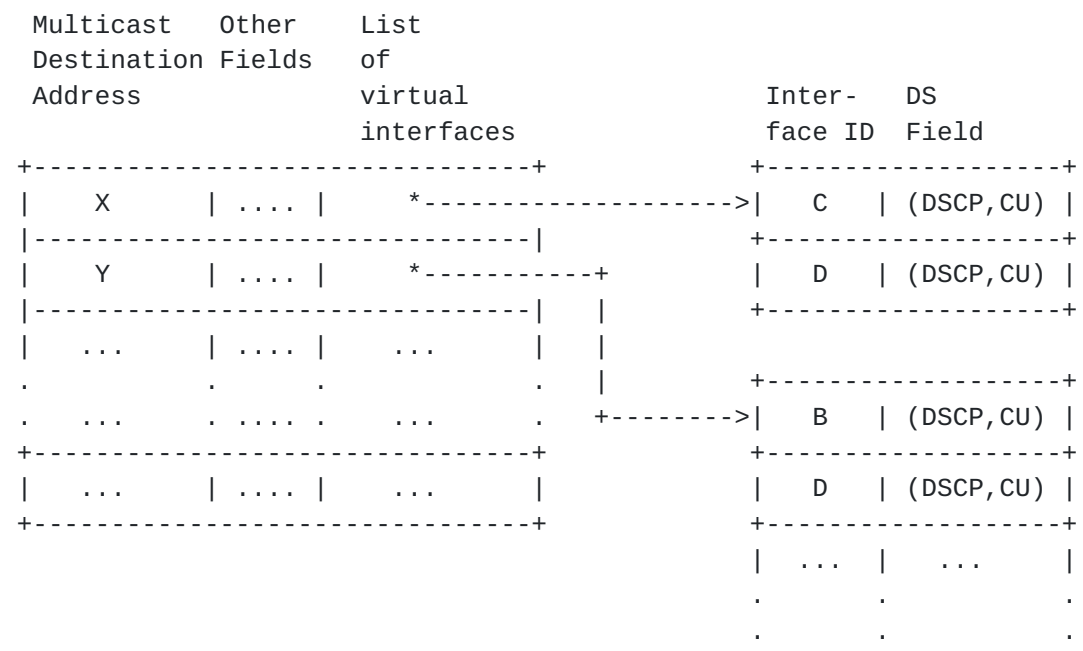


Figure 8: Multicast routing table with additional fields for DSCP values

8 Proof of the Neglected Reservation Subtree Problem

In the following, it is shown that the NRS problem actually exists and occurs in reality. Hence, the problem and its solution was investigated using a standard Linux Kernel (v2.4.18) and the Linux-based implementation KIDS [11].

Furthermore, the proposed solution for the NRS problem has been implemented by enhancing the multicast routing table as well as the multicast routing behavior in the Linux kernel. In the following section, the modifications are briefly described.

Additional measurements with the simulation model simulatedKIDS [12] will be presented in section 9. They show the effects of the NRS problem in more detailed and also the behavior of the BAs using or not using the Limited Effort PHB for re-marking.

8.1 Implementation of the proposed solution

As described in section 3.1, the proposed solution for avoiding the NRS Problem is an extension of each routing table entry in every Multicast router by one byte. In the Linux OS the multicast routing table is implemented by the "Multicast Forwarding Cache (MFC)". The MFC is a hash table consisting of an "mfc-cache"-entry for each combination of the following three parameters: sender's IP address, multicast group address and incoming interface.

The routing information in a "mfc-cache"-entry is kept in an array of TTLs for each virtual interface. When the TTL is zero, a packet matching to this "mfc-cache"-entry will not be forwarded on this virtual interface. Otherwise, if the TTL is less than the packet's TTL, the latter will be forwarded on the interface with a decreased TTL.

In order to set an appropriate codepoint if bandwidth is allocated on an outgoing link, we added a second array of bytes -- similar to the TTL array -- for specifying the codepoint that should be used on a particular virtual interface. The first six bits of the byte contain the DSCP that should be used and the seventh bit indicates, whether the original codepoint in the packet has to be changed to the specified one (=0) or has to be left unchanged (=1). The default entry of the codepoint byte is zero, so initially all packets will be re-marked to the default DSCP.

Furthermore, we modified the multicast forwarding code for considering this information while replicating multicast packets. To change an "mfc-cache"-entry we implemented a daemon for exchanging the control information with a management entity (e.g., a bandwidth broker). Currently, the daemon uses a proprietary protocol, but it is planned to migrate to the COPS protocol (RFC 2748).

8.2 Test Environment and Execution

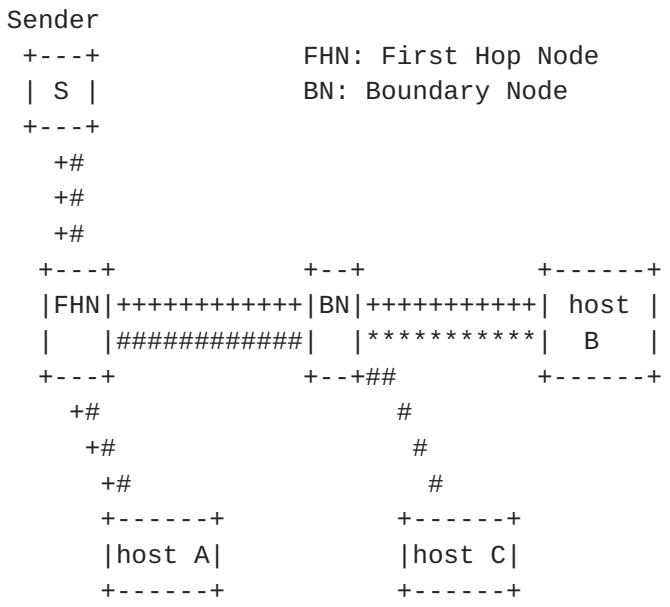


Figure 8.1: Evaluation of NRS-Problem described in Figure 3

In order to prove case 1 of the NRS problem, as described in section 2.1, a testbed shown in Figure 8.1 was built. It is a reduced version of the network shown in Figure 5 and consists of two DS-capable node, an ingress boundary node and an egress boundary node. The absence of interior nodes does not have any effects on to the proof of the described problem.

The testbed comprises of two Personal Computers (Pentium III at 450 Mhz, 128 MB Ram, 3 network cards Intel eepro100) used as DiffServ nodes, as well as one sender and three receiver systems (also PCs). On the routers KIDS has been installed and a mrouted (Multicast Routing Daemon) was used to perform multicast routing. The network was completely built of separate 10BaseT Ethernet segments in full-duplex mode. In [11] we evaluated the performance of the software routers and found out that even a PC at 200Mhz had no problem to handle up to 10Mbps DS traffic on each link. Therefore, the presented measurements are not a result of performance bottlenecks caused by these software routers.

The sender generated two shaped UDP traffic flows of 500kbps (packets of 1.000 byte constant size) each and sends them to

multicast group 1 (233.1.1.1) and 2 (233.2.2.2). In both

measurements receiver A had a reservation along the path to the sender for each flow, receiver B has reserved for flow 1 and C for flow 2. Therefore, two static profiles are installed in the ingress boundary node with 500kbps EF bandwidth and a token bucket size of 10.000byte for each flow.

In the egress boundary node one profile has been installed for the output link to host B and one related for the output link to host C. Each of them permits up to 500kbps Expedited Forwarding, but only the aggregate of Expedited Forwarding traffic carried on the outgoing link is considered.

In measurement 1 the hosts A and B joined to group 1 and A, B and C joined to group 2. Those joins are using a reservation for the group towards the sender. Only the join of host B to group 2 has no admitted reservation. As described in section 2.1 this will cause the NRS problem (case 1). Metering and policing mechanisms in the egress boundary node throttle down the EF aggregate to the reserved 500kbps, no depending on whether individual flows have reserved or not.

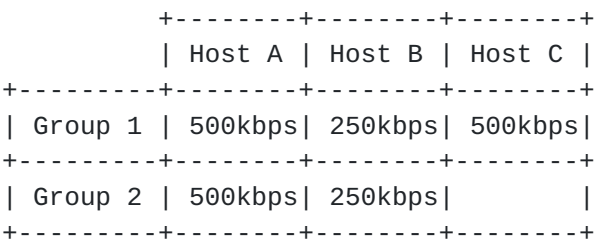


Figure 8.2: Results of measurement 1 (without the proposed solution): Average bandwidth of each flow.
--> Flows of group 1 and 2 on the link to host B share the reserved aggregate of group 1.

Figure 8.2 shows the obtained results. Host A and C received their flows without any interference. But host B received data from group 1 only with half of the reserved bandwidth, so one half of the packets have been discarded. Figure 8.2 also shows that receiver B got the total amount of bandwidth for group 1 and 2, that is exactly the reserved 500kbps. Flow 2 got Expedited Forwarding without actually having reserved any bandwidth and additionally violated the guarantee of group 1 on that link.

For measurement 2 the previously presented solution (cf. section 3.1) has been installed in the boundary node. Now it checks during duplicating the packets, whether the codepoint has to be changed to

Best-Effort (or Limited Effort) or whether it can be just

duplicated. In this measurement it changed the codepoint for group 2 on the link to Host B to Best-Effort.

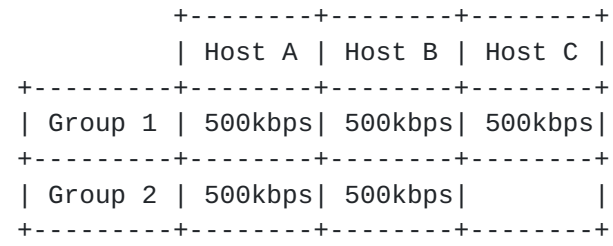


Figure 8.3: Results of measurement 1 (with the proposed solution): Average bandwidth of each flow.
--> Flow of group 1 on the link to host B gets the reserved bandwidth of group 1.
The flow of group 2 has been re-marked to Best-Effort.

Results of this measurement are presented in Figure 8.3. Each host received its flows with the reserved bandwidth and without any packet loss. Packets from group 2 are re-marked in the boundary node so that they have been treated as best-effort traffic. In this case, they got the same bandwidth as the Expedited Forwarding flow, because there was not enough other traffic on the link present, and thus no need to discard packets.

The above measurements confirm that the Neglected Reservation Subtree problem is to be taken seriously and that the presented solution will solve it.

9 Simulative Study of the NRS Problem and Limited Effort PHB

This section shows some results from a simulative study which shows the correctness of the proposed solution and the effect of re-marking the responsible flow to Limited Effort. A proof of the NRS problem has also been given in section 8 and in [13]. This section shows the benefit for the default Best Effort traffic when Limited Effort is used for re-marking instead of Best Effort. The results strongly motivate the use of Limited Effort.

9.1 Simulation Scenario

In the following scenario the boundary nodes had a link speed of 10 Mpbs and Interior Routers had a link speed of 12 Mbps. In boundary nodes a 5 Mbps aggregate for EF has been reserved.

When Limited Effort was used, LE got 10% capacity (0.5Mpbs) from the original BE aggregate and BE 90% (4.5Mbps) of the original BE

aggregate capacity. The bandwidth between LE and BE is shared by using WFQ scheduling.

The following topology was used, where Sx is a sender, BRx a boundary node, IRx an interior node and Dx a destination/receiver.

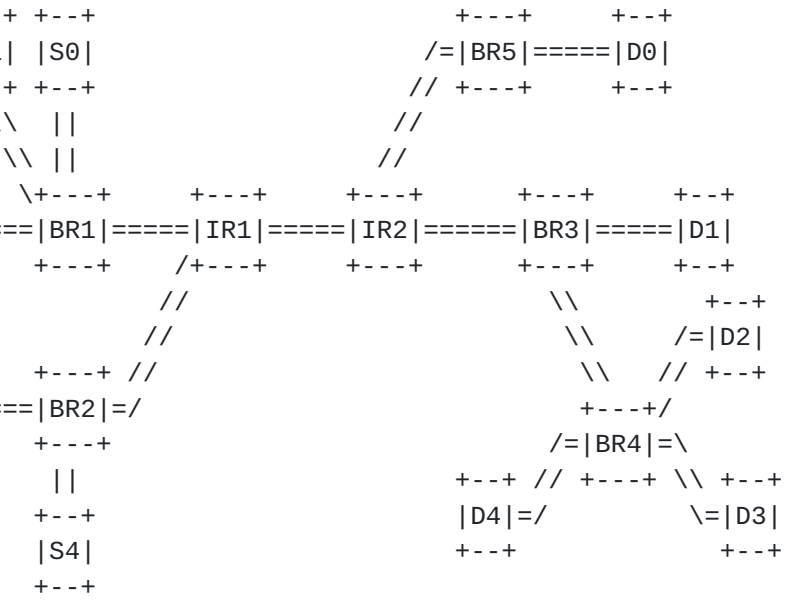


Figure 9.1: Simulation Topology

The following table shows the flows in the simulation runs, e.g., EF0 is sent from Sender S0 to Destination D0 with a rate of 4 Mbps using an EF reservation.

In the presented cases (I to IV) different amounts of BE traffic were used to show the effects of Limited Effort in different cases. The intention of these four cases is described after the table.

In all simulation models EF sources generated constant rate traffic with constant packet sizes using UDP.
The BE sources also generated constant rate traffic, where BE0 used UDP and BE1 used TCP as transport protocol.

Flow	Source	Dest.	Case I	Case II	Case III	Case IV
EF0	S0	D0	4 Mbps	4 Mbps	4 Mbps	4 Mbps
EF1	S1	D1	2 Mbps	2 Mbps	2 Mbps	2 Mbps
EF2	S2	D2	5 Mbps	5 Mbps	5 Mbps	5 Mbps
BE0	S3	D3	1 Mbps	2.25 Mbps	0.75 Mbps	3.75 Mbps
BE1	S4	D4	4 Mbps	2.25 Mbps	0.75 Mbps	3.75 Mbps

Table 9.1: Direction, amount and Codepoint of flows in the four simulation cases (case I to IV)

The four cases (I to IV) used in the simulation runs had the following characteristics:

Case I: In this scenario the BE sources sent together exactly 5 Mbps so there is no congestion in the BE queue.

Case II: BE is sending less than 5 Mbps, so there is space available in the BE queue for re-marked traffic. BE0 and BE1 are sending together 4.5 Mbps, which is exactly the share of BE, when LE is used. So when multicast packets are re-marked to LE because of the NRS problem, then LE should get 0.5 Mbps and BE 4.5 Mbps, which is still enough for BE0 and BE1. LE should not show a greedy behavior and should not use resources from BE.

Case III: In this case BE is very low. BE0 and BE1 use together only 1.5 Mbps. So when LE is used, it should be able to use the unused bandwidth resources from BE.

Case IV: BE0 and BE1 send together 7.5 Mbps so there is congestion in the BE queue. In this case LE should get 0.5 Mbps (not more and not less).

In each scenario loss rate and throughput of the considered flows and aggregates have been metered.

9.2 Simulation Results for different router types

9.2.1 Interior Node

When the branching point of a newly added multicast subtree is located in an interior node the NRS problem can occur as described in section 2.1 (Case 2).

In the simulation runs presented in the following four subsections D3 joins to the multicast group of sender S0 without making any reservation or resource allocation. Consequently a new branch is added to the existing multicast tree. The branching point issued by the join of D3 is located in IR2. On the link to BR3 no bandwidth was allocated for the new flow (EF0).

The metered throughput of flows on the link between IR2 and BR3 in the four different cases is shown in the following four subsections. The situation before the new receiver joins is shown in the second column. The situation after the join without the proposed solution is shown in column three. The fourth column presents the results when the proposed solution of section 3.1 is used and the responsible flow is re-marked to LE.

9.2.1.1 Case I:

	before join	after join (no re-marking)	after join, (re-marking to LE)
achieved through-put	EF0: ---	EF0: 4.007 Mbps	LE0: 0.504 Mbps
	EF1: 2.001 Mbps	EF1: 2.003 Mbps	EF1: 2.000 Mbps
	EF2: 5.002 Mbps	EF2: 5.009 Mbps	EF2: 5.000 Mbps
	BE0: 1.000 Mbps	BE0: 0.601 Mbps	BE0: 1.000 Mbps
	BE1: 4.000 Mbps	BE1: 0.399 Mbps	BE1: 3.499 Mbps
BA through-put	EF: 7.003 Mbps	EF: 11.019 Mbps	EF: 7.000 Mbps
	BE: 5.000 Mbps	BE: 1.000 Mbps	BE: 4.499 Mbps
	LE: ---	LE: ---	LE: 0.504 Mbps
packet loss rate	EF0: ---	EF0: 0 %	LE0: 87.4 %
	EF1: 0 %	EF1: 0 %	EF1: 0 %
	EF2: 0 %	EF2: 0 %	EF2: 0 %
	BE0: 0 %	BE0: 34.8 %	BE0: 0 %
	BE1: 0 %	BE1: 59.1 %	BE1: 0 %

(*) EF0 is re-marked to LE and signed as LE0

9.2.1.2 Case II:

	before join	after join (no re-marking)	after join, (re-marking to LE)
achieved through-put	EF0: ---	EF0: 4.003 Mbps	LE0: 0.500 Mbps
	EF1: 2.000 Mbps	EF1: 2.001 Mbps	EF1: 2.001 Mbps
	EF2: 5.002 Mbps	EF2: 5.005 Mbps	EF2: 5.002 Mbps
	BE0: 2.248 Mbps	BE0: 0.941 Mbps	BE0: 2.253 Mbps
	BE1: 2.252 Mbps	BE1: 0.069 Mbps	BE1: 2.247 Mbps
BA through-put	EF: 7.002 Mbps	EF: 11.009 Mbps	EF: 7.003 Mbps.
	BE: 4.500 Mbps	BE: 1.010 Mbps	BE: 4.500 Mbps
	LE: ---	LE: ---	LE: 0.500 Mbps
packet loss rate	EF0: ---	EF0: 0 %	LE0: 87.4 %
	EF1: 0 %	EF1: 0 %	EF1: 0 %
	EF2: 0 %	EF2: 0 %	EF2: 0 %
	BE0: 0 %	BE0: 58.0 %	BE0: 0 %
	BE1: 0 %	BE1: 57.1 %	BE1: 0 %

(*) EF0 is re-marked to LE and signed as LE0

9.2.1.3 Case III:

	before join	after join (no re-marking)	after join, (re-marking to LE)
achieved through-put	EF0: ---	EF0: 3.998 Mbps	LE0: 3.502 Mbps
	EF1: 2.000 Mbps	EF1: 2.001 Mbps	EF1: 2.001 Mbps
	EF2: 5.000 Mbps	EF2: 5.002 Mbps	EF2: 5.003 Mbps
	BE0: 0.749 Mbps	BE0: 0.572 Mbps	BE0: 0.748 Mbps
	BE1: 0.749 Mbps	BE1: 0.429 Mbps	BE1: 0.748 Mbps
BA through-put	EF: 7.000 Mbps	EF: 11.001 Mbps	EF: 7.004 Mbps
	BE: 1.498 Mbps	BE: 1.001 Mbps	BE: 1.496 Mbps
	LE: ---	LE: ---	LE: 3.502 Mbps
packet loss rate	EF0: ---	EF0: 0 %	LE0: 12.5 %
	EF1: 0 %	EF1: 0 %	EF1: 0 %
	EF2: 0 %	EF2: 0 %	EF2: 0 %
	BE0: 0 %	BE0: 19.7 %	BE0: 0 %
	BE1: 0 %	BE1: 32.6 %	BE1: 0 %

(*) EF0 is re-marked to LE and signed as LE0

9.2.1.4 Case IV:

	before join	after join (no re-marking)	after join, (re-marking to LE)
achieved-through-put	EF0: ---	EF0: 4.001 Mbps	LE0: 0.500 Mbps
	EF1: 2.018 Mbps	EF1: 2.000 Mbps	EF1: 2.003 Mbps
	EF2: 5.005 Mbps	EF2: 5.001 Mbps	EF2: 5.007 Mbps
	BE0: 2.825 Mbps	BE0: 1.000 Mbps	BE0: 3.425 Mbps
	BE1: 2.232 Mbps	BE1: ---	BE1: 1.074 Mbps
BA-through-put	EF: 7.023 Mbps	EF: 11.002 Mbps	EF: 7.010 Mbps
	BE: 5.057 Mbps	BE: 1.000 Mbps	BE: 4.499 Mbps
	LE: ---	LE: ---	LE: 0.500 Mbps
packet loss rate	EF0: ---	EF0: 0 %	LE0: 75.0 %
	EF1: 0 %	EF1: 0 %	EF1: 0 %
	EF2: 0 %	EF2: 0 %	EF2: 0 %
	BE0: 23.9 %	BE0: 73.3 %	BE0: 0 %
	BE1: 41.5 %	BE1: ---	BE1: 0 %

(*) EF0 is re-marked to LE and signed as LE0

NOTE: BE1 has undefined throughput and loss in situation "after join (no re-marking)", because TCP is going into retransmission back-off timer phase and closes the connection after 512 seconds.

9.2.2 Boundary Node

When the branching point of a newly added multicast subtree is located in a boundary node the NRS problem can occur as described in section 2.1 (Case 1).

In the simulation runs presented in the following four subsections D3 joins to the multicast group of sender S1 without making any reservation or resource allocation. Consequently, a new branch is added to the existing multicast tree. The branching point issued by the join of D3 is located in BR3. On the link to BR4 no bandwidth was allocated for the new flow (EF1).

The metered throughput of the flows on the link between BR3 and BR4 in the four different cases is shown in the following four subsections. The situation before the new receiver joins is shown in the second column. The situation after the join but without the proposed solution is shown in column three. The fourth column presents results when the proposed solution of section 3.1 is used and the responsible flow is re-marked to LE.

9.2.2.1 Case I:

	before join	after join (no re-marking)	after join, (re-marking to LE)
achieved through-put	EF0: ---	EF0: ---	EF0: ---
	EF1: ---	EF1: 1.489 Mbps	LE1: 0.504 Mbps
	EF2: 5.002 Mbps	EF2: 3.512 Mbps	EF2: 5.002 Mbps
	BE0: 1.000 Mbps	BE0: 1.000 Mbps	BE0: 1.004 Mbps
	BE1: 4.000 Mbps	BE1: 4.002 Mbps	BE1: 3.493 Mbps
BA through-put	EF: 5.002 Mbps	EF: 5.001 Mbps	EF: 5.002 Mbps
	BE: 5.000 Mbps	BE: 5.002 Mbps	BE: 4.497 Mbps
	LE: ---	LE: ---	LE: 0.504 Mbps
packet loss rate	EF0: ---	EF0: ---	EF0: ---
	EF1: ---	EF1: 25.6 %	LE1: 73.4 %
	EF2: 0 %	EF2: 29.7 %	EF2: 0 %
	BE0: 0 %	BE0: 0 %	BE0: 0 %
	BE1: 0 %	BE1: 0 %	BE1: 0 %

(*) EF1 is re-marked to LE and signed as LE1

9.2.2.2 Case II:

	before join	after join (no re-marking)	after join, (re-marking to LE)
achieved through-put	EF0: ---	EF0: ---	EF0: ---
	EF1: ---	EF1: 1.520 Mbps	LE1: 0.504 Mbps
	EF2: 5.003 Mbps	EF2: 3.482 Mbps	EF2: 5.002 Mbps
	BE0: 2.249 Mbps	BE0: 2.249 Mbps	BE0: 2.245 Mbps
	BE1: 2.252 Mbps	BE1: 2.252 Mbps	BE1: 2.252 Mbps
BA through-put	EF: 5.003 Mbps	EF: 5.002 Mbps	EF: 5.002 Mbps
	BE: 4.501 Mbps	BE: 4.501 Mbps	BE: 4.497 Mbps
	LE: ---	LE: ---	LE: 0.504 Mbps
packet loss rate	EF0: ---	EF0: ---	EF0: ---
	EF1: ---	EF1: 24.0 %	LE1: 74.8 %
	EF2: 0 %	EF2: 30.4 %	EF2: 0 %
	BE0: 0 %	BE0: 0 %	BE0: 0 %
	BE1: 0 %	BE1: 0 %	BE1: 0 %

(*) EF1 is re-marked to LE and signed as LE1

9.2.2.3 Case III:

	before join	after join (no re-marking)	after join, (re-marking to LE)
achieved through-put	EF0: ---	EF0: ---	EF0: ---
	EF1: ---	EF1: 1.084 Mbps	LE1: 2.000 Mbps
	EF2: 5.001 Mbps	EF2: 3.919 Mbps	EF2: 5.000 Mbps
	BE0: 0.749 Mbps	BE0: 0.752 Mbps	BE0: 0.750 Mbps
	BE1: 0.749 Mbps	BE1: 0.748 Mbps	BE1: 0.750 Mbps
BA through-put	EF: 5.001 Mbps	EF: 5.003 Mbps	EF: 5.000 Mbps
	BE: 1.498 Mbps	BE: 1.500 Mbps	BE: 1.500 Mbps
	LE: ---	LE: ---	LE: 2.000 Mbps
packet loss rate	EF0: ---	EF0: ---	EF0: ---
	EF1: ---	EF1: 45.7 %	LE1: 0 %
	EF2: 0 %	EF2: 21.7 %	EF2: 0 %
	BE0: 0 %	BE0: 0 %	BE0: 0 %
	BE1: 0 %	BE1: 0 %	BE1: 0 %

(*) EF1 is re-marked to LE and signed as LE1

9.2.2.4 Case IV:

	before join	after join (no re-marking)	after join, (re-marking to LE)
achieved through-put	EF0: ---	EF0: ---	EF0: ---
	EF1: ---	EF1: 1.201 Mbps	LE1: 0.500 Mbps
	EF2: 5.048 Mbps	EF2: 3.803 Mbps	EF2: 5.004 Mbps
	BE0: 2.638 Mbps	BE0: 2.535 Mbps	BE0: 3.473 Mbps
	BE1: 2.379 Mbps	BE1: 2.536 Mbps	BE1: 1.031 Mbps
BA through-put	EF: 5.048 Mbps	EF: 5.004 Mbps	EF: 5.004 Mbps
	BE: 5.017 Mbps	BE: 5.071 Mbps	BE: 4.504 Mbps
	LE: ---	LE: ---	LE: 0.500 Mbps
packet loss rate	EF0: ---	EF0: ---	EF0: ---
	EF1: ---	EF1: 40.0 %	LE1: 68.6 %
	EF2: 0 %	EF2: 23.0 %	EF2: 0 %
	BE0: 30.3 %	BE0: 32.1 %	BE0: 0 %
	BE1: 33.3 %	BE1: 32.7 %	BE1: 0 %

(*) EF1 is re-marked to LE and signed as LE1

10 References

Normative References

[1] F. Baker, D. Black, S. Blake, and K. Nichols. Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers. RFC 2474, Dec. 1998.

[2] S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang, and W. Weiss. An Architecture for Differentiated Services. RFC 2475, Dec. 1998.

Informative References

[3] K. Nichols, B. Carpenter. Definition of Differentiated Services Per Domain Behaviors and Rules for their Specification. RFC 3086, Apr. 2001.

[4] R. Braden, S. Berson, S. Herzog, S. Jamin, and L. Zhang. Resource ReSerVation Protocol (RSVP) -- Version 1. RFC 2205, Sept. 1997.

[5] Y. Bernet, Format of the RSVP DCLASS Object, RFC 2996, November 2000.

[6] D. Waitzman, C. Partridge, and S. Deering. Distance Vector Multicast Routing Protocol. RFC 1075, Nov. 1988.

[7] D. Estrin, D. Farinacci, A. Helmy, D. Thaler, S. Deering, M. Handley, V. Jacobson, C. gung Liu, P. Sharma, and L. Wei. Protocol Independent Multicast-Sparse Mode (PIM-SM): Protocol Specification. RFC 2362, June 1998.

[8] A. Adams, J. Nicholas, W. Siadak. Protocol Independent Multicast - Dense Mode (PIM-DM) Protocol Specification (Revised). Internet-Draft draft-ietf-pim-dm-new-v2-03.txt, February 2003, work in progress.

[9] F. Baker, J. Heinanen, W. Weiss, and J. Wroclawski. Assured Forwarding PHB Group. RFC 2597, June 1999.

[10] Y. Bernet, S. Blake, D. Grossman, A. Smith. An Informal Management Model for DiffServ Routers. RFC 3290, May 2002

[11] R. Bless, K. Wehrle. Evaluation of Differentiated Services using an Implementation und Linux, Proceedings of the Intern. Workshop on Quality of Service (IWQOS'99), London, 1999

[12] K. Wehrle, J. Reber, V. Kahmann. A simulation suite for Internet nodes with the ability to integrate arbitrary Quality of Service behavior, Proceedings of Communication Networks And Distributed Systems Modeling And Simulation Conference (CNDS 2001), Phoenix (AZ), January 2001

[13] R. Bless, K. Wehrle. Group Communication in Differentiated Services Networks, Internet QoS for the Global Computing 2001 (IQ 2001), IEEE International Symposium on Cluster Computing and the Grid, May 2001, Brisbane, Australia, IEEE Press

11 Acknowledgements

The authors wish to thank Mark Handley and Bill Fenner for their valuable comments to this document. Special thanks go to Milena Neumann for her extensive efforts in performing the simulations. We would also like to thank the KIDS simulation team [12].

Funding for the RFC Editor function is currently provided by the Internet Society.

12 Authors' Addresses

Comments and questions related to this document can be addressed to one of the authors listed below.

Roland Bless
Institute of Telematics
Universitaet Karlsruhe (TH)
Zirkel 2
76128 Karlsruhe, Germany
Phone: +49 721 608 6413
Email: bless@tm.uka.de
URI: http://www.tm.uka.de/~bless

Klaus Wehrle
Inst. of Telematics, Univ. of Karlsruhe
Zirkel 2, 76128 Karlsruhe, Germany &
Intern. Computer Science Institute (ICSI)
1947 Center Str, Berkeley, CA 94704, USA
Email: klaus@wehrle.com
URI: http://www.icsi.berkeley.edu/~wehrle

13 IPR Notice

The IETF takes no position regarding the validity or scope of any intellectual property or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; neither does it represent that it has made any effort to identify any such rights. Information on the IETF's procedures with respect to rights in standards-track and standards-related documentation can be found in BCP-11. Copies of claims of rights made available for publication and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementors or users of this specification can be obtained from the IETF Secretariat.

The IETF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights which may cover technology that may be required to practice this standard. Please address the information to the IETF Executive Director.

14 Copyright Notice