Network Working Group
Request for Comments: 3353
Category: Informational

D. Ooms
Alcatel
B. Sales
Alcatel
W. Livens
Colt Telecom
A. Acharya
IBM
F. Griffoul
Ulticom
F. Ansari
Bell Labs

August 2002

Overview of IP Multicast in a Multi-Protocol Label Switching (MPLS) Environment

Status of this Memo

Copyright Notice

Abstract

This document offers a framework for IP multicast deployment in an MPLS environment. Issues arising when MPLS techniques are applied to IP multicast are overviewed. The pros and cons of existing IP multicast routing protocols in the context of MPLS are described and the relation to the different trigger methods and label distribution modes are discussed. The consequences of various layer 2 (L2) technologies are listed. Both point-to-point and multi-access networks are considered.

Table of Contents

Table of Abbreviations

ATM	Asynchronous Transfer Node
CBT	Core Based Tree
CoS	Class of Service
DLCI	Data Link Connection Identifier
DRrecv	Designated Router of the receiver
DRsend	Designated Router of the sender
DVMRP	Distant Vector Multicast Routing Protocol
FR	Frame Relay
IGMP	Internet Group Management Protocol
IP	Internet Protocol
L2	layer 2 (e.g. ATM, Frame Relay)
L3	layer 3 (e.g. IP)
LSP	Label Switched Path
LSR	Label Switching Router
LSRd	Downstream LSR
LSRu	Upstream LSR
MOSPF	Multicast OSPF
mp2mp	multipoint-to-multipoint
MRT	Multicast Routing Table
p2mp	point-to-multipoint
PIM-DM	Protocol Independent Multicast-Dense Mode
PIM-SM	Protocol Independent Multicast-Sparse Mode
QoS	Quality of Service
RP	Rendezvous Point
RPT-bit	RP Tree bit [DEER]
RSVP	Resource reSerVation Protocol
SPT-bit	Shortest Path Tree [DEER]
SSM	Source Specific Multicast
TCP	Transmission Control Protocol
UDP	User Datagram Protocol
VC	Virtual Circuit
VCI	Virtual Circuit Identifier
VP	Virtual Path
VPI	Virtual Path Identifier

1. Introduction

In an MPLS cloud the routes are determined by a L3 routing protocol. These routes can then be mapped onto L2 paths to enhance network performance. Besides this, MPLS offers a vehicle for enhanced network services such as QoS/CoS, traffic engineering, etc.

Current unicast routing protocols generate a same (optimal) shortest path in steady state for a certain (source, destination) pair. Remark that unicast protocols can behave slightly different with regard to equal cost paths.

For multicast, the optimal solution (minimum cost to interconnect N nodes) would impose a Steiner tree computation. Unfortunately, no multicast routing protocol today is able to maintain such an optimal tree. Different multicast protocols will therefore, in general, generate different trees.

The discussion is focused on intra-domain multicast routing protocols. Aspects of inter-domain routing are beyond the scope of this document.

2. Layer 2 Characteristics

Although MPLS is multiprotocol both at L3 and at L2, in practice IP is the only considered L3 protocol. MPLS can run on top of several L2 technologies (PPP/Sonet, Ethernet, ATM, FR, ...).

When label switching is mapped on L2 switching capabilities (e.g. VPI/VCI is used as label), attention is mainly focused on the mapping to ATM [DAVI]. ATM offers high switching capacities and QoS awareness, but in the context of MPLS it poses several limitations which are described in [DAVI]. Similar considerations are made for Frame Relay on L2 in [CONT]. The limitations can be summarized as:

- Limited Label Space: either the standardized or the implemented number of bits available for a label can be small (e.g. VPI/VCI space, DLCI space), limiting the number of LSPs that can be established.

- Merging: some L2 technologies or implementations of these technologies do not support multipoint-to-point and/or multipoint-to-multipoint 'connections', obstructing the merging of LSPs.

- TTL: L2 technologies do not support a 'TTL-decrement' function.

All three limitations can impact the implementation of multicast in MPLS as will be described in this document.

When native MPLS is deployed the above limitations vanish. Moreover on PPP and Ethernet links the same label can be used at the same time for a unicast and a multicast LSP because different EtherTypes for MPLS unicast and multicast are defined [ROSE].

3. Taxonomy of IP Multicast Routing Protocols in the Context of MPLS

At the moment, an abundance of IP multicast routing protocols is being proposed and developed. All these protocols have different characteristics (scalability, computational complexity, latency, control message overhead, tree type, etc...). It is not the purpose of this document to give a complete taxonomy of IP multicast routing protocols, only their characteristics relevant to the MPLS technology will be addressed.

The following characteristics are considered:

- Aggregation
- Flood & Prune
- Source/Shared trees
- Co-existence of Source and Shared Trees
- Uni/Bi-directional shared trees
- Encapsulated multicast data
- Loop-free-ness

The discussion of these characteristics will not lead to the selection of one superior multicast routing protocol. It is not impossible that different IP multicast routing protocols will be deployed in the Internet.

3.1. Aggregation

In unicast different destination addresses are aggregated to one entry in the routing table, yielding one FEC and one LSP.

The granularity of multicast streams is (*, G) for a shared tree and (S, G) for a source tree, S being the source address and G the multicast group address. Aggregation of multicast trees with different multicast 'destination' addresses on one LSP is a subject for further study.

3.2. Flood & Prune

To establish a multicast tree some IP multicast routing protocols (e.g. DVMRP, PIM-DM) flood the network with multicast data. The branches can then be pruned by nodes which do not want to receive the data of the specific multicast group. This process is repeated periodically.

Flood & Prune multicast routing protocols have some characteristics which significantly differ from unicast routing protocols:

a) Volatile. Due to the Flood & Prune nature of the protocol, very volatile tree structures are generated. Solutions to map a dynamic L3 p2mp tree to a L2 p2mp LSP need to be efficient in terms of signaling overhead and LSP setup time. The volatile L2 LSP will consume a lot of labels throughout the network, which is a disadvantage when label space is limited.

b) Traffic-driven. The router only creates state for a certain group when data arrives for that group. Routers also independently decide to remove state when an inactivity timer expires.

   - Thus LSPs can not be pre-established as is usually done in unicast. To minimize the time between traffic arrival and LSP establishment a fast LSP setup method is favorable.

   - Since creation and deletion of a L3 route at each node is triggered by traffic, this suggests that the LSP associated with the route be setup and torn down in a traffic-driven manner as well.

   - If an LSR does not support L3 forwarding this traffic-driven nature even requires that the upstream LSR takes the initiative to create an LSP (Upstream Unsolicited or Downstream on Demand label advertisement).

3.3. Source/Shared Trees

IP multicast routing protocols create either source trees (S, G), i.e. a tree per source (S) and per multicast group (G), or shared trees (*, G), i.e. one tree per multicast group (Figure 1).

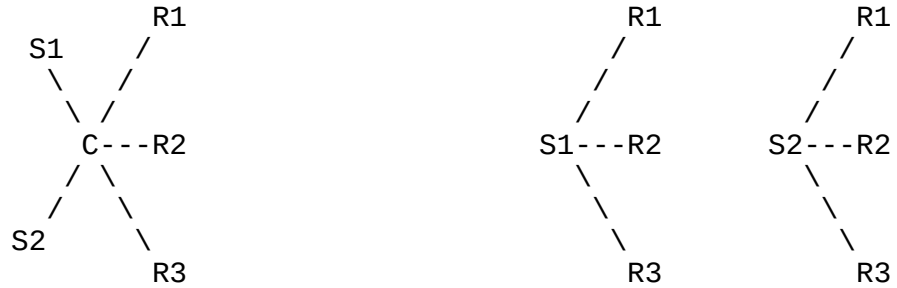

Figure 1. Shared tree and Source trees

The advantage of using shared trees, when label switching is applied, is that shared trees consume less labels than source trees (1 label per group versus 1 label per source and per group).

However, mapping a shared tree end-to-end on L2 implies setting up multipoint-to-multipoint (mp2mp) LSPs. The problem of implementing mp2mp LSPs boils down to the merging problem discussed earlier.

Note that in practice shared trees are often only used to discover new sources of the group and a switchover to a source tree is made at very low bitrates.

3.4. Co-existence of Source and Shared Trees

Some protocols support both source and shared trees (e.g. PIM-SM) and one router can maintain both (*, G) and (S, G) state for the same group G. Two cases of state co-existence are described below. Assume topologies with senders Si and receivers Ri. RP is the Rendezvous Point. Ni are LSRs. The numbers are the interface numbers, "Reg" is the Register interface. All IGMP and PIM Join/Prune messages are shown in the figures. It is also indicated whether the RPT-bit is set for the (S, G) state.

1) Figure 2 shows a switchover from shared to source tree. Assume that the shortest path from R1 to RP is via N1-N2-N5. N1, the Designated Router of receiver R1 (DRrecv), decides to initiate a source tree for source S1. After the arrival of data via the source tree in N2, N2 will send a prune to N5 for source S1. State co-existence occurs in the node where the overlap of shared and source tree starts (N2) and in the node where S1 does not need forwarding on the shared tree anymore (N5).

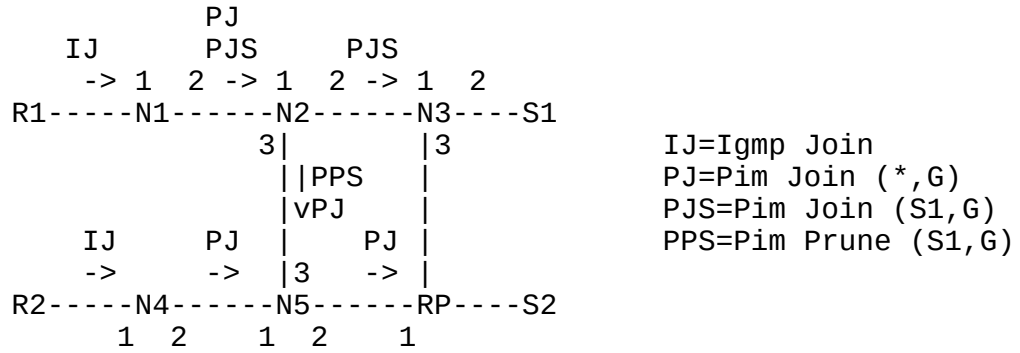

Figure 2

The multicast routing states created in the Multicast Routing Table (MRT) are:

```
in RP: (*,G):Reg->1   (i.e. incoming itf=Reg; outgoing itf=1)
in N1: (*,G):2->1
in N2: (*,G):3->1
       (S1,G):2->1
in N3: (S1,G):2->Reg,1
in N4: (*,G):2->1
in N5: (*,G):2->1,3
       (S1,G)RPT-bit:2->1
```

2) Figure 3 shows that even without a switchover, state co-existence can occur. Multicast traffic from a sender will create (S, G) state in the Designated Router of the sender (DRsend; N3 in Figure 3 is the DRsend of S). Each node on a shared-tree has (*, G) state. Thus an on-tree DRsend has both (*, G) and (S, G) state. If the DRsend is on-tree it will also send a prune for S towards the RP, creating (S, G) state in all nodes until the first router which has a branch (N1 and N2 in Figure 3).

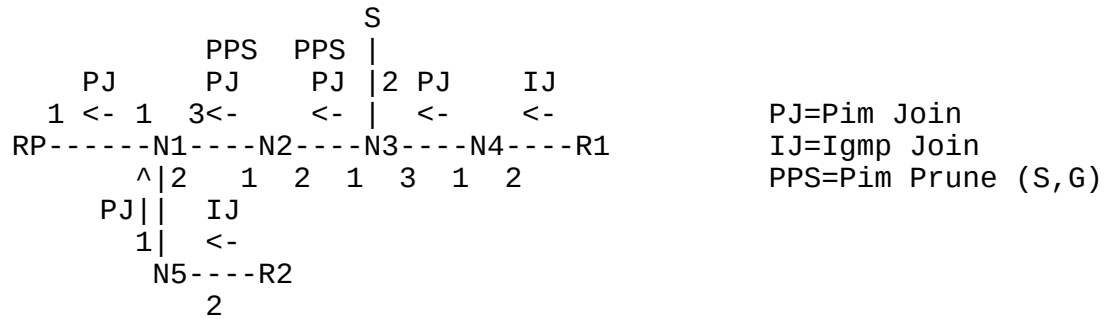

Figure 3

The multicast routing states created in the MRT are:

```
in RP: (*,G):Reg->1   (i.e. incoming itf=Reg; outgoing itf=1)
in N1: (*,G):1->2,3
       (S,G)RPT-bit:1->2
in N2: (*,G):1->2
       (S,G)RPT-bit:1->none
in N3: (*,G):1->3
       (S,G):2->Reg,3
in N4: (*,G):1->2
in N5: (*,G):1->2
```

In the examples one can observe that two types of state co-existence occur:

1) (S, G) with RPT-bit not set (N2 in Figure 2, N3 in Figure 3). The (*, G) and (S, G) state have different incoming interfaces, but some common outgoing interfaces. It is possible that the traffic of S arrives on both the (*, G) and (S, G) interfaces. In normal L3 forwarding the (S, G)SPT-bit entry prohibits the forwarding of the traffic from S arriving on the (*, G) incoming interface. The traffic of S can only temporarily arrive on the incoming interfaces of both the (*, G) and (S, G) entries (until N5 in Figure 2 and N1 in Figure 3 have processed the prune messages). To avoid the temporary forwarding of duplicate packets L3 forwarding can be applied in this type of node. If one does not mind the temporary duplicate packets L2 forwarding can be applied. In this case the (*, G) and (S, G) streams have to be merged into the (*, G) LSP on their common outgoing interfaces.

2) (S, G) with RPT-bit set (N5 in Figure 2, N1 in Figure 3). The (*, G) and (S, G) state have the same incoming interface. The (S, G) traffic must be extracted from the (*, G) stream. In MPLS this state co-existence can be handled in several ways. Four approaches to this problem will be described:

   a) A first method to handle this state co-existence is to terminate the LSPs and forward all traffic of this group at L3. However a return to L3 can be avoided in case a (S, G) entry without an outgoing interface is added to the MRT (N2 in Figure 3). This entry will only receive traffic temporarily. In this particular case one could ignore the (S, G) state and maintain the existing (*, G) LSP, the disadvantage being duplicate traffic for a very short time.

   b) A second approach is to assign source specific labels on the nodes of the shared tree. Multiple labels will be associated with one (*, G) entry, corresponding to one label per active source. Since the nodes only know which sources are active when traffic from these sources arrives, the LSPs cannot be pre-established and a fast LSP setup method is favorable.

   c) A third way is that only source trees are labelswitched and that traffic on the shared tree is always forwarded at L3. This assumes that the shared tree is only used as a way for the receivers to find out who the sources are. By configuring a low bitrate switchover threshold, one can ensure that the receivers switchover to source trees very quickly.

d) In the fourth approach, an LSR which has (S, G) RPT-bit state with a non-null oif, advertises a label for (S, G) to the upstream LSR and this label advertisement is then propagated by each upstream LSR towards the RP. In this way a dedicated LSP is created for (S, G) traffic from the RP to the LSR with the (S, G) RPT-bit state. In the latter LSR, the (S, G) LSP is merged onto the (*, G) LSP for the appropriate outgoing interfaces. This ensures that (S, G) packets traveling on the shared tree do not make it past any LSR which has pruned S.

3.5. Uni/Bi-directional Shared Trees

Bidirectional shared trees (e.g. CBT [BALL]) have the disadvantage of creating a lot of merging points (M) in the nodes (N) of the shared tree. Figure 4 shows these merging points resulting from 2 senders S1 and S2 on a bidirectional tree.

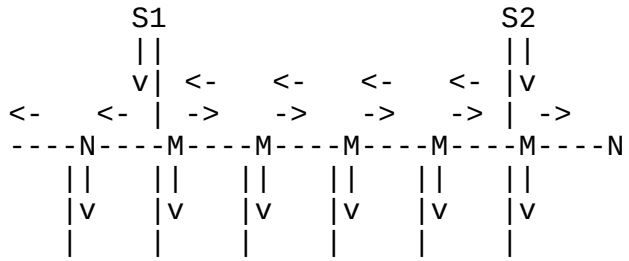

Figure 4.
Multicast traffic flows from 2 senders on a bidirectional tree

In Figure 5 the same situation for unidirectional shared trees is depicted. In this case the data of the senders is tunneled towards the root node R, yielding only a single merging point, namely the root of the shared tree itself.

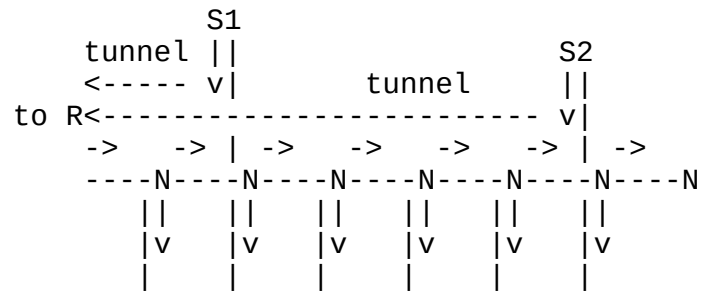

Figure 5.
Multicast traffic flows from 2 senders on a unidirectional tree

3.6. Encapsulated Multicast Data

Sources of unidirectional shared trees and non-member sources of bidirectional shared trees encapsulate the data towards the root node. The data is then decapsulated in the root node. The encapsulation and decapsulation of multicast data are L3 processes.

Thus in case of encapsulation/decapsulation a path can never be mapped onto an end-to-end LSP: the traffic can not be forwarded on L2 on the Register interface of the DRsend (encapsulation), nor can it cross the root (decapsulation) at L2.

Remarks:

1) If the LSR supports mixed L2/L3 forwarding (section 4), the (S, G) traffic in DRsend can still be forwarded at L2 on all outgoing interfaces other than the Register interface.

2) The encapsulated traffic can also benefit from MPLS by label switching the tunnels.

3) If the root node decides to join the source (to avoid encapsulation/decapsulation), an end-to-end (S, G) LSP can be constructed.

3.7. Loop-free-ness

Multicast routing protocols which depend on a unicast routing protocol suffer from the same transient loops as the unicast protocols do, however the effect of loops will be much worse in the case of multicast. The reason being, each time a multicast packet goes around a loop, copies of the packet may be emitted from the loop if branches exist in the loop.

Currently loop detection is a configurable option in LDP and a decision on the mechanism for loop prevention is postponed.

3.8. Mapping of Characteristics on Existing Protocols

The above characteristics are summarized in Table 1 for a non-exhaustive list of existing IP multicast routing protocols: DVMRP [PUSA], MOSPF [MOY], CBT [BALL], PIM-DM [ADAM], PIM-SM [DEER], SSM [HOLB], SM [PERL].

	DVMRP	MOSPF	CBT	PIM-DM	PIM-SM	SSM	SM
Aggregation	no	no	no	no	no	no	no
Flood & Prune	yes	no	no	yes	no	no	option
Tree Type	source	source	shared	source	both	source	shared
State Co-existence	no	no	no	no	yes	no	no
Uni/Bi-directional	N/A	N/A	bi	N/A	uni	uni	bi
Encapsulation	no	no	yes	no	yes	no	yes
Loop Free	no	no	no	no	no	no	no

Table 1. Taxonomy of IP Multicast Routing Protocols

From Table 1 one can derive e.g. that DVMRP will consume a lot of labels when the Flood & Prune L3 tree is mapped onto a L2 tree. Furthermore since DVMRP uses source trees it experiences no merging problem when label switching is applied. The table can be interpreted in the same way for the other protocols.

4. Mixed L2/L3 Forwarding in a Single Node

Since unicast traffic has one incoming and one outgoing interface the traffic is either forwarded at L2 OR at L3 (Figure 6). Because multicast traffic can be forwarded to more than one outgoing interface one can consider the case that traffic to some branches is forwarded on L2 and to other branches on L3 (Figure 7).

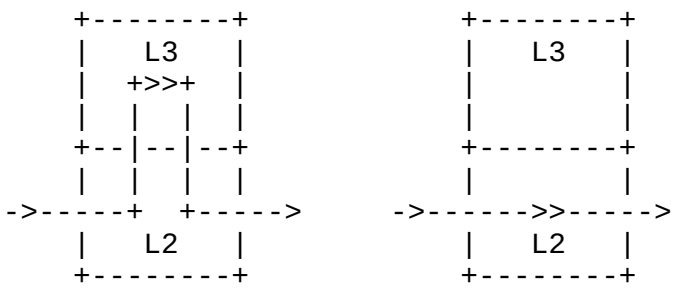

Figure 6. Unicast forwarding on resp. L3 or L2

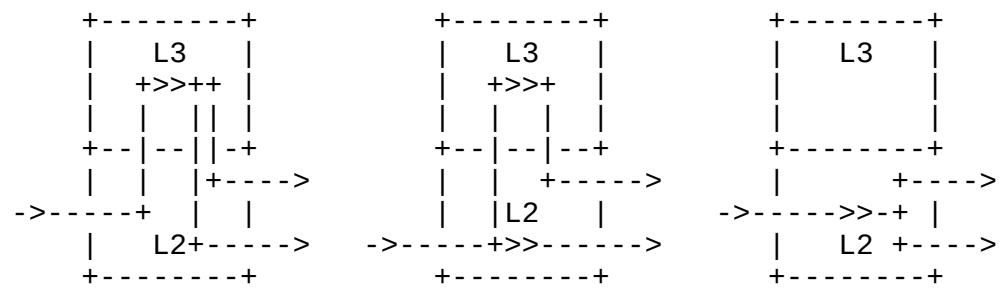

Figure 7. Multicast forwarding on resp. L3, mixed L2/L3 or L2

Nodes that support this 'mixed L2/L3 forwarding' feature allow splitting of a multicast tree into branches in which some are forwarded at L3 while others are switched at L2.

The L3 forwarding has to take care that the traffic is not forwarded on those branches that already get their traffic on L2. This can be accomplished by e.g. providing an extra bit in the Multicast Routing Table.

Although the mixed L2/L3 forwarding requires processing of the traffic at L3, the load on the L3 forwarding engine is generally less than in a pure L3 node.

Supporting this 'mixed L2/L3 forwarding' feature has the following advantages:

a) Assume LSR A (Figure 8) is an MPLS edge node for the branch towards LSR B and an MPLS core node for the branch towards LSR C. The mixed L2/L3 forwarding allows that the branch towards C is not disturbed by a return to L3 in LSR A.

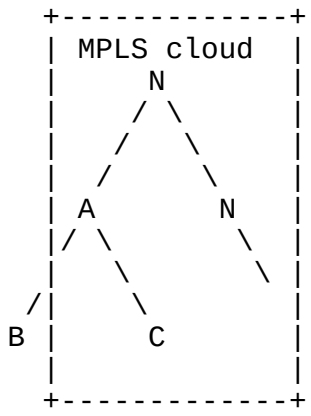

Figure 8. Mixed L2/L3 forwarding in node A

b) Enables the use of the traffic driven trigger with the Downstream Unsolicited or Upstream on Demand label distribution mode, as explained in section 5.3.1.

5. Taxonomy of IP Multicast LSP Triggers

The creation of an LSP for multicast streams can be triggered by different events, which can be mapped on the well known categories of 'request driven', 'topology driven' and 'traffic driven'.

a) Request driven: intercept the sending or receiving of control messages (e.g. multicast routing messages, resource reservation messages).

b) Topology driven: map the L3 tree, which is available in the Multicast Routing Table, to a L2 tree. The mapping is done even if there is no traffic.

c) Traffic driven: the L3 tree is mapped onto a L2 tree when data arrives on the tree.

5.1. Request Driven

5.1.1. General

The establishment of LSPs can be triggered by the interception of outgoing (requiring that the label is requested by the downstream LSR) or incoming (requiring that the label is requested by the upstream LSR) control messages. Figure 9 illustrates these two cases.

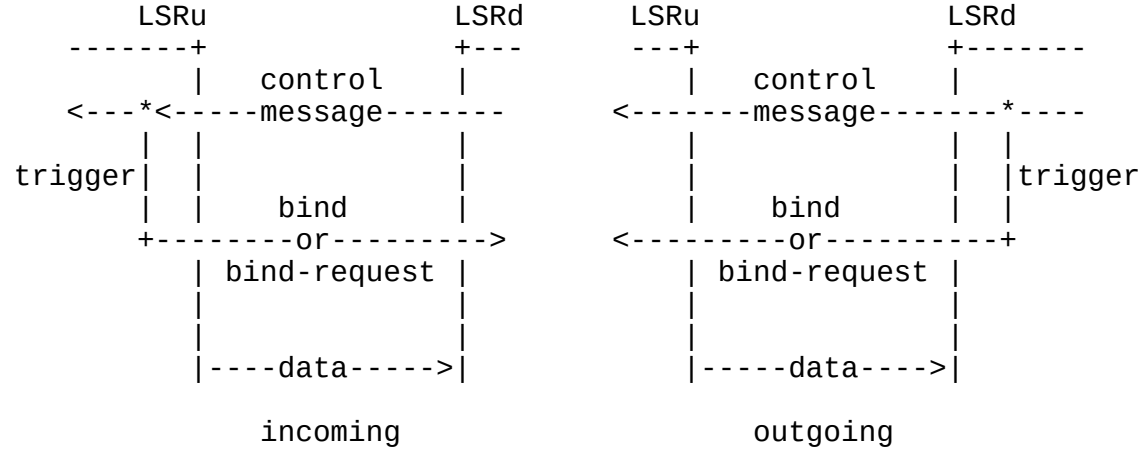

Figure 9. Request driven trigger
(interception of resp. incoming and outgoing control messages)

The downstream LSR (LSRd) sends a control message to the upstream LSR (LSRu). In the case that incoming control messages are intercepted and the MPLS module in LSRu decides to establish an LSP, it will send an LDP bind (Upstream Unsolicited mode) or an LDP bind request (Downstream on Demand mode) to LSRd.

Currently, for multicast, we can identify two important types of control messages: the multicast routing messages and the resource reservation messages.

5.1.2. Multicast Routing Messages

In principle, this mechanism can only be used by IP multicast routing protocols which use explicit signaling: e.g. the Join messages in PIM-SM or CBT. Remark that DVMRP and PIM-DM can be adapted to support this type of trigger [FARI], however, at the cost of modifying the IP multicast routing protocol itself!

IP multicast routing messages can create both hard states (e.g. CBT Join + CBT Join-Ack) and soft states (e.g. PIM-SM Joins are sent periodically). The latter generates more control traffic for tree maintenance and thus requires more processing in the MPLS module.

Triggers based on the multicast routing protocol messages have the disadvantage that the 'routing calculations' performed by the multicast routing daemon to determine the Multicast Routing Table are repeated by the MPLS module. The former determines the tree that will be used at L3, the latter calculates an identical tree to be used by L2. Since the same task is performed twice, it is better to create the multicast LSP on the basis of information extracted from the Multicast Routing Table itself (see section 5.2 and 5.3). The routing calculations become more complex for protocols which support a switch-over from a (*, G) tree to a (S, G) tree because more messages have to be interpreted.

When a host has a point-to-point connection to the first router one could create 'LSPs up to the end-user' by intercepting not only the multicast routing messages but the IGMP Join/Prune messages ([FENN]) as well.

5.1.3. Resource Reservation Messages

As is the case for unicast the RSVP Resv message can be used as a trigger to establish LSPs. A source of a multicast group will send an RSVP Path message down the tree, the receivers can then reply with an RSVP Resv message. RSVP scales equally well for multicast as it does for unicast because:

a) RSVP Resv messages can merge.

b) RSVP Resv messages are only sent up to the first branch which made the required reservation.

5.2. Topology Driven

The Multicast Routing Table (MRT) is maintained by the IP multicast routing protocol daemon. The MPLS module maps this L3 tree topology information to L2 p2mp LSPs.

The MPLS module can poll the MRT to extract the tree topologies. Alternatively, the multicast daemon can be modified to notify the MPLS module directly of any change to the MRT.

The disadvantage of this method is that labels are consumed even when no traffic exists.

5.3. Traffic Driven

5.3.1. General

A traffic driven trigger method will only construct LSPs for trees which carry traffic. It consumes less labels than the topology driven method, as labels are only allocated when there is traffic on the multicast tree.

If the mixed L2/L3 forwarding capability (see section 4) is not supported, the traffic driven trigger requires a label distribution mode in which the label is requested by the LSRu (Downstream on Demand or Upstream Unsolicited mode). In Figure 10, suppose an LSP for a certain group exists to LSRd1 and another LSRd2 wants to join the tree. In order for LSRd2 to initiate a trigger, it must already receive the traffic from the tree. This can be either at L2 or at L3. The former case is a chicken and egg problem. The latter case requires a mixed L2/L3 forwarding capability in LSRu to add the L3 branch.

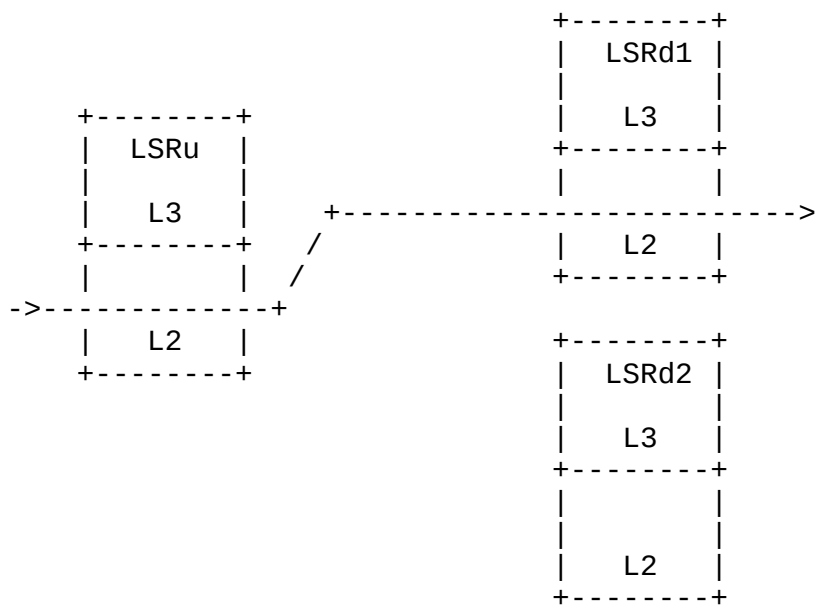

Figure 10. The LSRu has to request the label.

5.3.2. An Implementation Example

To illustrate that by choosing an appropriate trigger one can conclude that MPLS multicast is independent of the deployed multicast routing protocol, the following implementation example is given.

Current implementations on Unix platforms of IP multicast routing protocols (DVMRP, PIM) have a Multicast Forwarding Cache (MFC). The MFC is a cached copy of the Multicast Routing Table. The MFC requests an entry for a certain multicast group when it experiences a 'cache miss' for an incoming multicast packet. The missing routing information is provided by the multicast daemon. If at a later point in time something changes to the route (outgoing interfaces added or removed), the multicast daemon will update the MFC.

The MFC is implemented as a common component (part of the kernel), which makes this trigger very attractive because it can be transparently used for any IP multicast routing protocol.

Entries in the MFC are removed when no traffic is received for this entry for a certain period of time. When label switching is applied to a certain MFC-entry, the L3 will not see any packets arriving anymore. To retain the normal MFC behavior, the L3 counters of the MFC need to be updated by L2 measurements.

5.4. Combinations of Triggers and Label Distribution Modes

Table 2 shows the valid combinations of label distribution modes and trigger types that were discussed in the previous sections. The (X) means that the combination is valid when the mixed L2/L3 forwarding feature is supported in the LSR.

	label requested by			
	LSRu		LSRd	
	upstream unsolicited	downstream on demand	downstream unsolicited	upstream on demand
Request Driven (incoming msg)	X	X		
Request Driven (outgoing msg)			X	X
Topology Driven	X	X	X	X
Traffic Driven	X	X	(X)	(X)

Table 2. Valid combinations of triggers and label distribution modes

6. Piggy-backing

In Figure 9 (outgoing case) one can observe that instead of sending 2 separate messages the label advertisement can be piggy-backed on the existing control messages. For multicast two piggy-back candidates exist:

a) Multicast routing messages: protocols such as PIM-SM and CBT have explicit Join messages which could carry the label mappings. This approach is described in [FARI]. When different multicast routing protocols are deployed, an extension to each of these protocols has to be defined.

b) RSVP Resv messages: a label mapping extension object for RSVP, also applicable to multicast, is proposed in [AWDU].

The pros and cons of piggy-backing on multicast routing messages will be described now.

Piggy-backing has following advantages:

a) If the label advertisement is piggy-backed on multicast routing messages, then the distribution of routes and the distribution of labels is tightly synchronized. This eliminates difficult corner cases such as "what do I do with a label if I don't (yet) have a routing table entry to attach it to?". It also minimizes the interval between the establishment of the multicast route and the mapping of a label to the route.

b) The number of control messages needed to support label advertisement beyond those needed to support the multicast routing itself is zero.

Following disadvantages of piggy-backing can be identified:

a) In dense-mode protocols there are no messages on which the label advertisement can be piggy-backed. [FARI] proposes to add periodic messages to dense-mode protocols for the purpose of label advertisement, which is a heavy impact on the multicast routing protocol and it eliminates the message conserving benefit of piggy-backing.

b) The second solution for the state co-existence problem (section 3.4) cannot be applied in combination with piggy-backing.

c) Piggy-backing requires extending the multicast routing protocol, and hence becomes less attractive if label advertisement needs to be supported for multiple multicast routing protocols. Especially when not only the label advertisement but also the other two LDP functions (discovery and adjacency) are piggy-backed.

d) Piggy-backing assumes the Downstream Unsolicited label distribution mode, this excludes a number of trigger methods (see Table 2).

e) LDP normally runs on top of TCP, assuring a reliable communication between peer nodes. Piggy-backed label advertisement often replaces the reliable communication with periodic soft-state label advertisements. Because of this periodic label advertisement the control traffic (in number of bytes) will increase.

f) If a VCID notification mechanism [NAGA] is required, the (in-band) notification can normally be done by sending the LDP bind through the newly established VC. This way only one message is required. This method cannot be combined with piggy-backing because the routing message is sent before the VC can be established. An extra handshake message is thus required, diminishing the benefit of piggy-backing.

So whether piggy-backing makes sense or not depends heavily on which and how many multicast routing protocols are deployed, whether LDP is already used for unicast, which trigger mechanism is used, ... Piggy-backing is just one possible component of an MPLS multicast solution.

7. Explicit Routing

Explicit routing for unicast refers to overriding the unicast routing table by using LSPs.

A first way to interpret "multicast explicit routing" is overriding the tree established by the multicast routing protocol by another LSP tree (e.g. a Steiner tree calculated by an off-line tool). In this interpretation the current 'shortest path' multicast routing protocol becomes obsolete and can be replaced by label advertisement messages that follow an explicit route (e.g. a branch of the Steiner tree).

A second way of interpreting "multicast explicit routing" is that the known multicast routing protocols are running, but that the messages generated by these protocols use explicit unicast routes (instead of the IGP shortest path routes) to construct trees.

8. QoS/CoS

8.1. DiffServ

The Differentiated Services approach can be applied to multicast as well. It introduces finer stream granularities (DiffServ Codepoint (DSCP) as an extra differentiator). A sender can construct one or more trees with different DSCPs.

These (S, G, DSCP) or (*, G, DSCP) trees can be mapped very easily onto LSPs when the traffic driven trigger is used. In this case one can create LSPs with different attributes for the various DSCPs. Note however that these LSPs still use the same route as long as the tree construction mechanism itself does not take the DSCP as an input.

8.2. IntServ and RSVP

RSVP can be used to setup multicast trees with QoS. An important multicast issue is the problem of how to map the 'heterogeneous receivers' paradigm onto L2 (remark that it is not solved in IP either). This subject is tackled in [CRAW]. Pragmatic approaches are the 'Limited Heterogeneity Model' which allows a best effort service and a single alternate QoS (e.g. a QoS proposed by the sender in a RSVP Path message) and the 'Homogeneous Model' which allows only a single QoS.

The first approach will construct full trees for each service class. The sender has to send its traffic twice across the network (e.g. 1 best-effort and 1 QoS tree). Both trees can be label switched.

The second approach constructs one tree and the best-effort users are connected to the QoS tree. If the branches created for best-effort users are not to be label switched, (thus carried by a hop-by-hop default LSP) the QoS multicast traffic has to be merged onto these default LSPs. This function can be provided by the 'mixed L2/L3 forwarding' feature described in section 4. If this is not available, merging is necessary to avoid a return to L3 in the QoS LSP.

The mapping of the IntServ service categories onto L2 for ATM service categories is studied in [GARR].

9. Multi-access Networks

Multicast MPLS on multi-access networks poses a special problem. An LSR that wants to join a group must always be ready to accept the label that is already assigned to the group LSP (to another downstream LSR on the link). This can be achieved in three ways:

1) Each LSR on the multi-access link memorizes all the advertised labels on the link, even if it has not received a join for the associated group. If an LSR is added to the multi-access link it has to retrieve this information from another LSR on the link or in case of soft state label advertisement it can wait a certain time before it can allocate labels itself. If LSRs allocate a label 'at the same moment' the LSR with the highest IP address could keep it, while the other LSRs withdraw the label.

2) Each LSR gets its own label range to allocate labels from. A mechanism for label partitioning is described in [FARI]. If an LSR is added to the multi-access link, the label ranges have to be negotiated again and possibly existing LSPs are torn down and are reconstructed with other labels.

3) Per multi-access link one LSR could be elected to be responsible for label allocation. When an LSR needs a label, it can request it from this Label Allocation LSR.

Unlike the unicast case, a multicast stream can have more than one downstream LSR which all have to use the same label. Two solutions for label advertisement can be thought of:

1) [FARI] proposes to multicast the label advertisements to all LSRs on the shared link. Since multicast is not reliable this requires periodic label advertisements, yielding label advertisement duplicates in time.

2) Another approach is that an LSR unicasts its label advertisements in a reliable way (TCP) to all other (or to all interested) LSRs on the shared link. In this approach the hard-state character of LDP can be maintained but the label advertisement is duplicated in space.

Since LSPs are only rewarding if they have a long lifetime and since the number of LSRs on a shared link is limited the second approach seems advantageous.

Another issue with multicast in multi-access networks is whether to use upstream or downstream label assignment. For multicast traffic, upstream label allocation is simpler since there can be only one upstream node per link that belongs to a multicast tree. This (upstream) node can assign a unique label for the FEC. With downstream allocation, there may be multiple downstream nodes for a given tree on a multi-access link; each node may propose a different label assignment for a FEC that would require some resolution process in order to come up with a single label per multicast FEC on the link.

Once a label has been assigned, it is possible that the label assigner leaves the tree. With downstream label assignment, this could happen when the label allocator leaves the group. With upstream assignment this could happen when the upstream LSR changes due to a unicast topology change.

10. More Issues

10.1. TTL Field

The TTL field in the IP header is typically used for loop detection. In IP multicast it is also used to limit the scope of the multicast packets by setting an appropriate TTL value.

Thus in LSRs that do not support a TTL decrement function (e.g. ATM LSR), the scope restriction function is affected. Suppose one could calculate in advance the number of hops an LSP traverses. In a unicast LSP the TTL value could then be decremented at the ingress or the egress node. For multicast all the branches of the tree can have different lengths so the TTL can only be decremented at the egress node, potentially wasting bandwidth if the TTL turns out to be zero or negative.

10.2. Independent vs. Ordered Label Distribution Control

Current Label Distribution Terminology is only defined for unicast. The following sections explore what this terminology might mean in a multicast context.

In Independent Control ([ANDE]) each LSR can take the initiative to do a label mapping. In Ordered Control ([ANDE]) an LSR only maps a label when it already received a label from its next-hop.

All the previously described trigger methods (section 5) combine with Independent Control. Note that if the request driven approach is used with Independent Control the label distribution still behaves as in Ordered Control: the control messages flow from the egress node upstream, imposing the same sequence to the label advertisement.

Ordered Control is not applicable for a traffic driven trigger in case the node does not support mixed L2/L3 forwarding. According to Table 2, this case implies that labels are requested by the upstream LSR. Suppose in Figure 11 that an LSP exists from S to R1 and a new branch must be added to R2. B will only accept a label on the A-B link if a label is already assigned on the B-C link. However, to establish a label on the B-C link, B must already receive traffic on the A-B link.

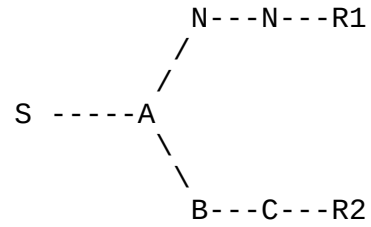

Figure 11.

10.3. Conservative vs. Liberal Label Retention Mode

In the Conservative Mode ([ANDE]) only the labels that are used for forwarding data (if the next-hop for the FEC is the LSR which advertised the label) are allocated and maintained. In the Liberal Mode labels are advertised and maintained to all neighbors. Liberal Mode does not make sense in multicast. Two reasons can be identified for this:

1) All LSRs have a route for each unicast FEC. This is not true for multicast FECs.

2) For multicast an LSR always knows to which neighbor to send the label request or the label map messages. In e.g. unicast Downstream Unsolicited mode (see below) the LSR does not know where to send the label mappings and thus has to send the mapping to all its neighbors. In this case supporting the Liberal Mode does not generate extra messages (it still requires extra state information and label space) and thus the threshold to support Liberal Mode could be considered lower.

Table 3 shows the cases where it is known by an LSR where to send its label requests.

	label requested by	
	LSRu	LSRd
unicast	Yes	No
multicast	Yes	Yes

Table 3. Does an LSR know where to send its label requests ?

For a unicast flow, an LSR can determine the next hop LSR, which is the one to send the request to in case of Upstream Unsolicited or Downstream on Demand mode. The LSR is however not able to find the previous hop. The previous hop is not necessarily the next hop towards the source, because the path from A to B is not necessarily the same as the path from B to A. Such a situation can occur as a result of asymmetric link measures or in the event that multiple equal cost paths exist [PAXS].

In the case of multicast, an LSR knows both the next hop(s) and the previous hop. Because multicast trees are constructed using the reverse shortest path method, the previous hop is always the next hop towards the source or towards the root of the tree.

10.4. Downstream vs. Upstream Label Allocation

The label can be allocated by either the downstream LSR (Downstream on Demand, Downstream Unsolicited) or the upstream LSR (Upstream on Demand, Upstream Unsolicited, implicit). The advantages of downstream label allocation are:

a) It is the same mode as for unicast LDP, thus eliminating the need to develop upstream label distribution procedures.

b) The same label can be kept when the upstream LSR changes due to a route change, which is an advantage on multi-access networks (see section 9).

c) Compatible with piggy-backing (especially the downstream distribution mode).

The advantages of upstream label allocation are:

a) Easier label allocation in multi-access networks (see section 9).

b) The same label can be kept when the downstream LSR (which would have been the label allocator in downstream mode in a multi-access network) leaves the group (see section 9).

c) The upstream and implicit distribution mode allow a faster LSP setup when the LSP is traffic triggered.

Whether to use upstream or downstream label distribution is outside the scope of this framework. The relative complexity between the necessary protocol extensions and the resolution mechanism needed, as well as the relative operational complexity, will influence which way to go.

10.5. Explicit vs. Implicit Label Distribution

Beside the explicit distribution modes (which use a signaling protocol), [ACHA] proposes an implicit label distribution method by using unknown labels. This method has all the advantages of the upstream label allocation method and is probably the fastest label advertisement method for traffic triggered LSPs.

Implicit label distribution is not applicable if the FEC-to-label binding has been advertised prior to traffic arrival, e.g. explicit routing (i.e. if all the information necessary to identify the FEC is not present in the packet).

Explicit distribution allows pre-establishment (before the arrival of data) of LSPs with topology or request driven triggers.

11. Security Considerations

In general, the use of multicast in an MPLS environment poses no extra security issues beyond the ones that already exist in multicast and MPLS protocols as such.

The protocols described in this document are however not suited to cross administrative boundaries.

When the multicast tree is determined by an existing multicast routing protocol (this is the assumption made in this document, except for the Explicit Routing section), clearly no additional security issues are introduced with respect to the shape of the tree (e.g. unauthorized joining, tapping, blackholing, injecting traffic, ...). These security issues should have been addressed in the specifications of the multicast routing protocols.

In the MPLS context it is possible that control messages trigger L2 resource allocations (e.g. LSPs). If security flaws would still be present in the existing protocols (which possibly are not too harmful in its original context) the abusive sending of such control messages can yield more severe DoS attacks.

In case of an "explicit routed" tree that is calculated centrally, sufficient authentication must be done on the control messages that set up the tree in the network nodes.

12. Acknowledgements

The authors would like to thank Eric Rosen, Piet Van Mieghem, Philip Dumortier, Hans De Neve, Jan Vanhoutte, Alex Mondrus and Gerard Gastaud for the fruitful discussions and/or their thorough revision of this document.

Informative References

[ACHA] A. Acharya, R. Dighe and F. Ansari, "IP Switching Over Fast ATM Cell Transport (IPSOFACTO) : Switching Multicast flows", IEEE Globecom '97.

[ADAM] A. Adams, J. Nicholas, W. Siadak, Protocol Independent Multicast Version 2 Dense Mode Specification", Work In Progress.

[ANDE] Andersson, L., Doolan, P., Feldman, N., Fredette, A. and R. Thomas, "LDP Specification", RFC 3036, January 2001.

[AWDU] Awduche, D., Berger, L., Gan, D., Li, T., Swallow, G. and V. Srinivasan, "RSVP-TE: Extensions to RSVP for LSP Tunnels", RFC 3209, December 2001.

[BALL] Ballardie, A., "Core Based Trees (CBT) Multicast Routing Architecture", RFC 2201, September 1997.

[CONT] Conta, D., Doolan, P. and A. Malis, "Use of Label Switching on Frame Relay Networks", RFC 3034, January 2001.

[CRAW] Crawley, E., Berger, L., Berson, S., Baker, F., Borden, M. and J. Krawczyk, "A Framework for Integrated Services and RSVP over ATM", RFC 2382, August 1998.

[DAVI] Davie, B., Lawrence, J., McCloghrie, K., Rekhter, Y., Rosen, E., Swallow, G. and P. Doolan, "MPLS using LDP and ATM VC switching", RFC 3035, January 2001.

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

[FARI] D. Farinacci, Y. Rekhter, E. Rosen and T. Qian, "Using PIM to Distribute MPLS Labels for Multicast Routes", Work In Progress.

[FENN] Fenner, W., "Internet Group Management Protocol, IGMP, Version 2", RFC 2236, November 1997.

[GARR] Garrett, M. and M. Borden, "Interoperation of Controlled-Load Service and Guaranteed Service with ATM", RFC 2381, August 1998.

[HOLB] H. Holbrook, B. Cain, "Source-Specific Multicast for IP", Work In Progress.

[MOY] Moy, J., "Multicast Extensions to OSPF", RFC 1584, March 1994.

[NAGA] Nagami, K., Demizu, N., Esaki, H., Katsube, Y. and P. Doolan, "VCID Notification over ATM link for LDP", RFC 3038, January 2001.

[PERL] R. Perlman, C-Y. Lee, A. Ballardie, J. Crowcroft, Z. Wang, T. Maufer, "Simple Multicast", Work In Progress.

[PUSA] T. Pusateri, "Distance Vector Multicast Routing Protocol", Work In Progress.

[PAXS] V. Paxson, "End-to-End Routing Behavior in the Internet", IEEE/ACM Transactions on Networking 5(5), pp. 601-615.

[ROSE] Rosen, E., Rekhter, Y., Tappan, D., Farinacci, D., Fedorkow, G., Li, T. and A. Conta, "MPLS Label Stack Encoding", RFC 3032, January 2001.

Authors Addresses

Dirk Ooms
Alcatel Corporate Research Center
Fr. Wellesplein 1, 2018 Antwerp, Belgium.
Phone : 32 3 2404732
Fax : 32 3 2409879
EMail: Dirk.Ooms@alcatel.be

Bernard Sales
Alcatel Corporate Research Center
Fr. Wellesplein 1, 2018 Antwerp, Belgium.
Phone : 32 3 2409574
EMail: Bernard.Sales@alcatel.be

Wim Livens
Colt Telecom
Zweefvliegtuigstraat 10, 1130 Brussels, Belgium
Phone : 32 2 7901705
Fax : 32 2 7901711
EMail: WLivens@colt-telecom.be

Arup Acharya
IBM TJ Watson Research Center
30 Saw Mill River Road, Hawthorne
NY 10532
Phone : 1 914 784 7481
EMail: arup@us.ibm.com

Frederic Griffoul
Ulticom, Inc.
Les Algorithmes, 2000 Route des Lucioles, BP29
06901 Sophia-Antipolis, FRANCE
EMail: griffoul@ulticom.com

Furquan Ansari
Bell Labs, Lucent Tech.
101 Crawfords Corner Rd., Holmdel, NJ 07733
Phone : 1 732 949 5249
Fax : 1 732 949 4556
EMail: furquan@dnrc.bell-labs.com

Full Copyright Statement

Acknowledgement

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