Internet Draft
October 2001
Expiration Date: April 2002

Paola Iovanna
Roberto Mameli
Giovanna Piantanida
Ericsson Lab Italy
Stefano Salsano
CoRiTeL/ Univ. of Rome "Tor Vergata"

Document: draft-iovanna-rsvp-mpls-flowspec-00.txt

Definition of the MPLS FlowSpec for RSVP-TE

Status of this Memo

Abstract

The support of QoS Differentiation in MPLS is based on the definition of two types of LSP, respectively E-LSP and L-LSP (see [11]). Both E-LSP and L-LSP can be established with bandwidth reservation. In this case bandwidth requirements for the LSP must be signaled at LSP establishment time, in order to perform admission control in the LSRs in the path. If RSVP-TE is used as setup protocol, it has to support the signaling of bandwidth requirements for the LSPs. In this document we extend the RSVP-TE capability by defining a new type of Sender_Tspec (and FlowSpec), explicitly designed to allow signaling of bandwidth requirements for E-LSP and L-LSP set up. This new type, called MPLS Sender_Tspec (MPLS FlowSpec), can include bandwidth requirements for different classes inside the LSP.

Table of Contents

Acronyms

The reader is assumed to be familiar with the terminology used in this document. A list of the acronyms used is reported below.

AF	Assured Forwarding
BA	Behavior Aggregate
BE	Best Effort
CLS	Controlled Load Service
CS	Class Selector
DF	Default Forwarding
DSCP	Differentiated Services Code Point
EF	Expedited Forwarding
EXP	EXPerimental (bits)
E-LSP	EXP-inferred-PSC LSP
GS	Guaranteed Service
LSP	Label Switched Path
LSR	Label Switched Router
L-LSP	Label-only-inferred-PSC LSP
MPLS	Multi Protocol Label Switching
OA	Ordered Aggregate
PHB	Per Hop Behavior
PHBID	PHB IDentification code
PSC	Packet Scheduling Class
RSVP-TE	RSVP Traffic Engineering extension
SLS	Service Level Specification
TE	Traffic Engineering
Tspec	Traffic specification

1. Introduction

MPLS support of Differentiated Services is based on the definition of two types of LSP (see [11]). They are commonly referred to as L-LSP (Label-only-inferred-PSC LSP) and E-LSP (EXP-inferred-PSC LSP). L-LSPs are used to carry traffic belonging to a single OA; they support a single PSC, explicitly signaled at LSP establishment time, while additional information useful for the packet treatment (e.g. the drop precedence) are conveyed by the EXP bits. E-LSPs, instead, are thought for the support of multiple OAs; in this case the EXP bits in the MPLS shim header indicate the PHB to be applied to the packet.

Both L-LSP and E-LSP can be established with bandwidth reservation. In the first case bandwidth requirements for the LSP are signaled at LSP establishment time; they are used by LSRs along the path to perform admission control over the provisioned DiffServ resources. The current specification of RSVP-TE states that bandwidth reservation for both E-LSP and L-LSP can be obtained by using IntServ Controlled Load Service, or possibly Guaranteed Service (see [9]); in this case the bandwidth requirements are signaled in the Sender_Tspec object of the PATH message (respectively FlowSpec of the RESV message).

This current approach for signaling bandwidth requirements is not completely satisfactory. First of all it is not correct from a conceptual point of view; in fact we use an IntServ Sender_Tspec (respectively IntServ FlowSpec) to signal bandwidth requirements for a scenario other than Integrated Service. Moreover in the case of E-LSP, the current specification does not allow to signal bandwidth requirements separately and independently for the different OAs carried by the E-LSP (bandwidth specification can only be given on aggregate basis).

There are some scenarios in which signaling and reservation of resources on a per-OA basis within an E-LSP could be desirable, e.g. to allow per-class resource management and admission control (see 7). This document defines a new type of Sender_Tspec (respectively FlowSpec), explicitly designed to allow signaling of bandwidth requirements for E-LSP and L-LSP set up. The new type is called MPLS Sender_Tspec (respectively MPLS FlowSpec). It supports the signaling of bandwidth requirements for a set of OAs independently.

The proposed solution applies to the MPLS support of Differentiated Service as defined in [11], but it can also be used for other scenarios where a MPLS network supports a set of traffic classes.

2. Definition of the MPLS Sender_Tspec (MPLS FlowSpec) object

In [13] the problem of signaling bandwidth requirements separately for the different OAs transported over a single E-LSP is examined. As stated there, at least three possible solutions exist, namely:

(1) Creation of a new object;

(2) Extension of the existing DiffServ object (defined in [11]);
(3) Extension of the existing Sender_Tspec (FlowSpec) to signal multiple FlowSpecs.

The first approach is the one followed in [13]; it consists in the creation of the new ELSP object at the purpose of signaling bandwidth requirements per OA. The second approach seems to be the least desirable, for several reasons. First of all, it would change the original meaning of the DiffServ object, on a second-hand it would be applicable only in the DiffServ over MPLS scenario and finally it would require the DiffServ object to be present in the RESV message.

In this document, instead, we propose to follow the third approach, i.e. to extend the existing Sender_Tspec (FlowSpec) with the introduction of a new C-Type. This is motivated by several reasons: first of all the introduction of a new Sender_Tspec (FlowSpec) type is more correct from a conceptual point of view. In fact it allows preserving the functionality of the Sender_Tspec (FlowSpec), since it keeps its original meaning of signaling traffic characteristics (with a finer granularity level of the reservation, i.e. on a per-OA basis).
In addition the new object type can be used for both E-LSP and L-LSP, and also for QoS over MPLS scenarios other than DiffServ. It can be designed in a flexible way, in order to allow signaling of bandwidth requirements in different ways, i.e. in accordance to different traffic models (possibly not only based on Tspec).
Moreover the coexistence between a new object for bandwidth reservation (as proposed by the first solution) and the current Sender_Tspec (FlowSpec) object could lead to conflicts.
Finally the introduction of a new C-Type does not raise particular

problems in terms of backward compatibility with existing implementations that do not support it.

The following paragraph describes the object in more details.

2.1 MPLS Sender_Tspec (MPLS FlowSpec)

In this section the structure of the new MPLS Sender_Tspec (FlowSpec) type is defined. It is exactly the same in both cases, with the obvious difference of the Class value.

Class = 9 (FlowSpec class)
Class = 12 (Sender_Tspec class)

C-type = 3 (MPLS Sender_Tspec/FlowSpec object)

The content and the encoding rules for this object are specified in the following.

```
   <MPLS Tspec > ::= <MPLS Tspec Header>
                     [<MPLS Tspec Body>]
```



```
<MPLS Tspec Body> ::= <List of MPLS Tspec SubObjects>

   <List of MPLS Tspect SubObjects ::= <MPLS Tspec SubObject>   |
                                       <MPLS Tspec SubObject>
                                       <List of MPLS Tspec SubObjects>

   <MPLS Tspec SubObject> :== <MPLS Tspec SubObject header>
                              <MPLS Tspec SubObject body>
```

```
   <MPLS Tspec Header>

                        1                   2                   3
    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |   Version     |     Flags     |   Reserved    | Num-of-SubObjs|
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Version: 8 bits
It specifies the version of the object and it is used for future extensions. Current value for Version is 1.

Flags: 8 bits
Can be used to define flags. Currently set to 0.

Reserved: 8 bits
Set to 0.

Num-of-SubObjs: 8 bits
Specifies the number of SubObjects in the the MPLS Tspec Body. In the case of absence of the MPLS Tspec Body, this is set to 0. This means that no bandwidth reservation is related to the LSP setup.

<MPLS Tspec object header>

```
                    1                   2                   3
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| SubObj-num   |           Reserved             |    Length     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

SubObj-num: 8 bits
Specifies the type of SubObject. Currently two types of SubObjects are defined:

SubObj-num = 1 : OA-Traffic Profile PHBID

SubObj-num = 2 : OA-Traffic Profile Simple Exp

OA-TP PHBID SubObject will be used in the MPLS/Diffserv scenario.

Reserved: 16 bits
Currently set to 0. It could contain SubObj specific information.

Length: 8 bits
Specifies the length of the SubObject body in 32bit words.

The <MPLS Tspec object body> is specific for each SubObj-num as follows:

<MPLS Tspec SubObject body> for OA-Traffic Profile PHBID

```
                     1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|             PHBID             |   TP-style    |    Length     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                   Traffic Profile Parameter                   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
//                        . . . . . . .                        //
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                   Traffic Profile Parameter                   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

<MPLS Tspec SubObject body> for OA-Traffic Profile Simple Exp

```
                     1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Exp |         Reserved        |   TP-style    |    Length     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                   Traffic Profile Parameter                   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
//                        . . . . . . .                        //
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                   Traffic Profile Parameter                   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Length: 8 bits
This field indicates the length of the <MPLS Tspec SubObject body> in 32 bit words.

PHBID: 16 bits
This field contains the Per Hop Behavior Identification Code, encoded as specified in [12].

Exp: 3 bits
This field contains the Exp field.

TP-Style: 8 bits
This field indicates the bandwidth requirement type. Different representations for the bandwidth requirements can be used,

according to the traffic models available. Currently two possible TP-Style values are defined, but other values could be introduced in the future if needed.

TP-style = 1 (Tspec parameters as defined for RSVP)

```
    0                   1                   2                   3
    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
  +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  |  Token Bucket Rate [r] (32-bit IEEE floating point number)    |
  +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  |  Token Bucket Size [b] (32-bit IEEE floating point number)    |
  +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  |  Peak Data Rate [p] (32-bit IEEE floating point number)       |
  +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  |  Minimum Policed Unit [m] (32-bit integer)                    |
  +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  |  Maximum Packet Size [M]  (32-bit integer)                    |
  +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

TP-style = 2 (Simple bandwidth parameter)

```
   0                   1                   2                   3
   0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
  +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  |    Bandwidth     [b]   (32-bit IEEE floating point number)    |
  +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

An example of a complete MPLS Sender Tspec object is given hereafter. It represents the bandwidth requirements for an E-LSP which supports two OA. Therefore, it includes the bandwidth requirements for two different PHBIDs, respectively expressed with a Token Bucket representation and with a Simple bandwidth parameter.

Example of a complete MPLS Tspec object

```
   0                   1                   2                   3
   0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
  +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  |          Length (bytes)       |  C-Num=12     |    C-Type=3   |
  +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  |   Version=1   |    Flags=0    |   Reserved    |Num-of-SubObj=2|
  +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  | SubObj-num=1  |           Reserved            |  Length=6     |
  +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  |          PHBID= xx            |  TP-style=1   |  Length=5     |
  +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  |  Token Bucket Rate [r] (32-bit IEEE floating point number)    |
  +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  |  Token Bucket Size [b] (32-bit IEEE floating point number)    |
  +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  |  Peak Data Rate [p] (32-bit IEEE floating point number)       |
  +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  |  Minimum Policed Unit [m] (32-bit integer)                    |
  +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  |  Maximum Packet Size [M]  (32-bit integer)                    |
  +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  | SubObj-num=1  |           Reserved            |  Length=2     |
  +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  |          PHBID= yy            |  TP-style=2   |  Length=1     |
  +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  |    Bandwidth     [b]   (32-bit IEEE floating point number)    |
  +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

3. Handling of the MPLS Sender_Tspec (MPLS FlowSpec) object in the DiffServ over MPLS scenario

This paragraph describes in details all the operations related to the new Sender_Tspec (FlowSpec) handling in the particular scenario of DiffServ over MPLS. In this case the object is intended to support signaling of bandwidth requirements on a per-OA basis for both E-LSP and L-LSP. However, for backward compatibility, Sender_Tspec (FlowSpec) objects with IntServ C-type can still be used in PATH (RESV) messages to signal bandwidth requirements for the entire flow, as specified in [9] and [11]. It is worth to observe that in the case of L-LSP the new MPLS C-type does not add new information with respect to the classical IntServ C-type; in fact there is no need to signal per-OA bandwidth requirements in such situations. Nevertheless the use of the MPLS Sender_Tspec (FlowSpec) is preferred for at least two reasons. First it is more correct from a conceptual point of view, as stated before, and second it allows signaling such bandwidth requirements in a more flexible way (e.g. representing only the bandwidth value instead of the

5-tuple Tspec).

A node that receives a PATH (RESV) message containing a MPLS Sender_Tspec (FlowSpec) but does not support it, must generate a PATH ERR (RESV ERR) message corresponding to ôUnknown object C-Typeö (Error Code = 14, see Appendix B of [1]).

To signal per-OA traffic requirements the ingress LSR must include a MPLS Sender_Tspec object in the PATH message. Correspondingly the receiver must include a MPLS FlowSpec object in the RESV message.

Each PSC signaled in the MPLS Sender_Tspec (MPLS FlowSpec) must be supported. In the case of E-LSP a corresponding set of EXP<->PHB mapping must exist. Of course this means that all the PHB associated to each PSC signaled in the MPLS Sender_Tspec (FlowSpec) must be supported, either pre-configured on every node or signaled by means of the DIFFSERV object ([11]). In the case of L-LSP the Encaps<->PHB mapping specified in the DIFFSERV object must coincide with the PSC signaled in the MPLS Sender_Tspec (FlowSpec) (Of course for a L-LSP this object must include only a single SubObject).

A node that receives a PATH (RESV) message with MPLS Sender_Tspec (FlowSpec) object containing one or more unsupported PSC must fail the reservation request and send a PATH_ERR (RESV ERR) message of the type ôDiff-Serv Errorö (Error code to be assigned by the IANA, see section **5.5 of [11]) and error value 2 (ôUnsupported PHBö) or 4 (ôUnsupported** PSCö).

A node that receives a RESV message with per-OA traffic profiles signaled in the MPLS FlowSpec object must perform Admission Control on every aggregate. If resources available for one or more OA signaled in MPLS FlowSpec are deemed insufficient, the node must fail the entire resource reservation. In this case it must send a RESV_ERR message with Error Code = 01 (ôAdmission Control Failureö) and Error Value of the form 0000cccccccccccc (ôAdmission Control Failure for an OAö, TBD).

If all the PHB/PSC signaled in the MPLS Sender_Tspec (MPLS FlowSpec) are supported and sufficient resources are available for each of them, then the node must carry out proper actions in order to let the reservation succeed (i.e. proper configuration of schedulers, classifiers and traffic conditioning elements).

A node must handle all the situations where a LSP cannot be accepted for other reason than the ones discussed in this section, in accordance to [1], [9] and [11].

4. Security Considerations

This document does not introduce additional security issues beyond those discussed in [1], [3], [7], [9] and [11]. As a consequence it may use the same mechanisms described in the references above.

5. IANA Considerations

This document defines a new object type with implications for IANA. A new C-type is defined for the MPLS Sender_Tspec (MPLS FlowSpec) object. The value (C-type = 3) has been chosen according to the First Come First Served policy defined in [14].

The new RSVP error code (ôDiffServ Errorö) should also be assigned from IANA (see section 5.5 of [11]). Finally a new error sub-code related to the ôAdmission Control Failureö error must be assigned.

6. Acknowledgements

Authors would like to thank all the people that participated to the revision of this document contributing with comments and suggestions.

7. Example scenario requiring per-OA admission control

There are many scenarios in which an MPLS network could support Differentiated Services by means of E-LSP. In such cases the provider would like to give service differentiation in its traffic engineered network without the complexity of managing different L-LSP for different services. Even if the provider is not interested in complex traffic engineering functionality (e.g. per class routing or per class protection/restoration), nevertheless it could be useful to manage resources on a per-OA basis, in order to allow better service provisioning and increased efficiency in network utilization.

For example, consider an ISP network that provides only two classes of service (e.g. Best Effort and Premium) using E-LSP. Assume that a generic node within the network is connected to a link with 100Mbit total bandwidth; assume it is able to provide 20Mbit of its bandwidth to Premium Service and the remaining 80Mbit to Best Effort traffic. Now imagine that the node receives two requests for E-LSP setup: E-LSP A and E-LSP B.

- E-LSP A requires 40Mbit of the link bandwidth (respectively 15Mbit for Premium Service and 25Mbit for Best Effort)
- E-LSP B requires 50Mbit of the link bandwidth (respectively 15Mbit for Premium Service and 35Mbit for Best Effort)

If the node does not support the MPLS Sender_Tspec (MPLS FlowSpec) extension described in this document, then it cannot perform per-OA admission control. In this case admission control is performed on aggregate basis. Since the total bandwidth requirement of the two E-LSPs is less than the link bandwidth, they are both accepted (90Mbit of link bandwidth, respectively 30Mbit for Premium Service and 60Mbit for Best Effort). Note however that the total amount of Premium traffic exceeds

the pre-configured limit (set to 20Mbit) and, as a consequence, Premium traffic on both E-LSP will be adversely affected.

In contrast, if the new object is supported and per-OA based Admission Control is performed, the node rejects one of the requests because it is able to control the bandwidth availability for each class of service. Upon rejection the network can react in various ways, from the simple choice of rerouting to more complex actions, such as resource provisioning and redistribution.

8. References

[1] R. Braden et al., ôResource Reservation Protocol (RSVP) û Version 1, Functional Specificationö, RFC 2205, September 1997
[2] J. Wroclawsky, ôThe use of RSVP with IETF Integrated Servicesö, RFC 2210, September 1997
[3] R. Braden et al., ôResource Reservation Protocol (RSVP) û Version 1 Message Processing Rulesö, RFC 2209, September 1997
[4] J. Wroclawsky, ôSpecification of the Controlled-Load Network Element Serviceö, RFC 2211, September 1997
[5] S. Shenker et al., ôSpecification of Guaranteed Quality of Serviceö, RFC 2212, September 1997
[6] Y. Bernet et al., ôSpecification of the Null Service Typeö, RFC 2997, November 2000
[7] E. Rosen et al., ôMultiprotocol Label Switching Architectureö, RFC 3031, January 2001

[8]   D. Awduche et al., ôRequirements for Traffic Engineering over MPLSö, RFC 2702, September 1999
[9]   D. Awduche et al., ôRSVP-TE: Extensions to RSVP for LSP Tunnelsö, draft-ietf-mpls-rsvp-lsp-tunnel-09.txt, February 2001
[10]   D. Awduche et al., ôApplicability Statement for Extensions to RSVP for LSP-Tunnelsö, draft-ietf-mpls-rsvp-tunnel-applicability-02.txt, April 2001
[11]   F. Le Faucheur et al., ôMPLS Support of Differentiated Servicesö, draft-ietf-mpls-diff-ext-09.txt, April 2001
[12]   D.Black et al., ôPer Hop Behavior Identification Codeö, RFC 3140, June 2001
[13]   S. Ganti et al., ôMPLS Support of Differentiated Services using E-LSPö, draft-ganti-mpls-diffserv-elsp-00.txt, April 2001
[14]   T. Narten et al., ôGuidelines for writing an IANA Considerations Section in RFCsö, RFC 2434, October 1998

9. AuthorsÆ Addresses

Paola Iovanna
Ericsson Lab Italy S.p.A.
Via Anagnina 203
00040 - Morena (Rome) û Italy
Tel.   +39 06 72582814

E-mail: paola.iovanna@eri.ericsson.se

Roberto Mameli
Ericsson Lab Italy S.p.A.
Via Anagnina 203
00040 - Morena (Rome) û Italy
Tel.   +39 06 72589119
E-mail: roberto.mameli@eri.ericsson.se

Giovanna Piantanida
Ericsson Lab Italy S.p.A.
Via Anagnina 203
00040 - Morena (Rome) û Italy
Tel.   +39 06 72583537
E-mail: giovanna.piantanida@eri.ericsson.se

Stefano Salsano
University of Rome Tor Vergata
CoRiTeL û Consorzio di Ricerca sulle Telecomunicazioni
c/o Ericsson - Via Anagnina 203

00040 - Morena (Rome) û Italy
Tel.   +39 06 72582540
E-mail: salsano@coritel.it