MPLS – Multiprotocol Label Switching

José Ruela, Manuel Ricardo

FEUP – Fac. Eng. Univ. Porto, Rua Dr. Roberto Frias, 4200-465 Porto, Portugal
INESC Porto, Campus da FEUP, Rua Dr. Roberto Frias, 378, 4200-465 Porto, Portugal
{jruela, mricardo}@inescporto.pt

Abstract

This article describes the main features of Multiprotocol Label Switching (MPLS), a standard architecture proposed by the IETF that integrates label swapping forwarding with network layer routing. The role of MPLS in overcoming limitations of overlay models, such as IP over ATM, and of conventional routing in IP networks, is discussed. The main concepts are introduced and the operation of MPLS is explained – classification of packets into Forward Equivalence Classes, label allocation and binding to routes, label distribution, set up of Label Switched Paths and route selection. Finally, support of traffic engineering and Quality of Service mechanisms in MPLS networks is analysed.

1. MPLS - rationale for a new routing and forwarding architecture

The success of the Internet is mainly due to its flexible architecture, built upon IP, the ubiquitous internetworking layer protocol.

IP networks offer unparalleled scalability and flexibility for the deployment of value-added services and are becoming increasingly attractive for carrying services with hard and soft real-time constraints. This requires extending the traditional best-effort model, designed from the outset for elastic data traffic, with mechanisms that provide differentiated and predictable Quality of Service (QoS) to a wide variety of applications with different requirements [1]. The Internet Engineering Task Force (IETF) has already specified the Integrated Services (IntServ) [2] and the Differentiated Services (DiffServ) [3] models with these goals in mind.

Carriers' requirements

With the increasing demand and explosive growth of the Internet, service providers require a dependable and controllable network infrastructure that can offer consistent performance. In many cases, the original router-based backbone networks evolved into a two level structure made up of a high speed core network interconnecting edge devices (IP routers) that, in turn, interface with access networks and provide common services to users, such as security, accounting, Virtual Private Networks (VPNs), web hosting, etc.

Management of such networks requires powerful traffic engineering techniques, that is, the capability of mapping flows into the physical network topology and evenly distributing traffic over the network links, to achieve an efficient utilization of network resources, avoid congestion and improve network performance.

Integration of IP and ATM

With the deployment of ATM (Asynchronous Transfer Mode) switches in many carriers' backbones, integrating IP and ATM appeared as a natural and attractive architectural choice to fulfil these goals, since it combined simple and robust routing techniques with a scalable, fast switching technology. Moreover, ATM includes provision for traffic management and differentiated transport services [4], which are essential in satisfying carriers' requirements.

However, this integration was not without problems. Initial solutions were based on overlay models, such as Classical IP over ATM (CLIP) [5], which are easy to deploy but have a number of serious shortcomings [6][7], especially in large networks where IP routers are interconnected by a full mesh of ATM connections, to avoid extra hops in the path across the backbone. In fact, in this architecture, two types of nodes (IP routers and ATM switches) run different signaling and routing protocols, on different address spaces, with the IP logical topology segregated from the ATM physical topology. This is inefficient and does not allow joint optimisation of resources. In addition, it poses scalability and stability problems, when the logical topology has to be reconfigured, due to the way routing updates are advertised between adjacent nodes by conventional IP routing protocols.

Multilayer switching and MPLS

To overcome these limitations, alternative architectures, based on the concept of multilayer switching, were investigated and proposed by manufacturers. Two such examples are IP Switching [8] and Tag Switching [9] but, in general, these solutions share a common principle – the separation of control and forwarding functions. The software based control component includes layer three routing and additional signaling protocols, while the hardware based forwarding component uses layer two label switching techniques. A mapping between routes and labels provides the glue between these components, required to build a multilayer switch.

The concept is not restricted to IP and ATM, but when the forwarding component is based on ATM, only the ATM switching fabric is retained (not the ATM-based control protocols).

In this context, the Multiprotocol Label Switching (MPLS) architecture [10] is the result of the current IETF work to standardise a solution that integrates label swapping forwarding with network layer routing, and that incorporates the basic principles and ideas proposed by manufacturers. Initial efforts are focused on IP, while ATM is a strong but not the unique choice for switching.

MPLS tries to solve most of the critical issues previously identified, with particular emphasis on scalability, fast packet forwarding and traffic engineering [11]. Although not dealing with specific QoS mechanisms (such as call admission, traffic shaping and policing, packet scheduling and discard policies), MPLS provides an appropriate framework for supporting a QoS architecture, as well.

2. Analysis of traditional packet switching techniques

In order to better understand how MPLS is built upon and extends the capabilities of existing technologies, it is useful to briefly review traditional connectionless and connection-oriented packet switching techniques.

In conventional IP networks, which operate in a connectionless (datagram) mode, packets are forwarded on a hop-by-hop basis, and each router along the path makes an independent forwarding decision. The next hop for a packet is selected based on the IP destination address and on forwarding information updated by means of routing protocols. The routing (forwarding) table is parsed in search of some address prefix that is the longest match for the packet's destination address.

This approach has a number of limitations, especially when we consider the role that IP networks are expected to play in the coming years.

First, forwarding decisions in each node are only based on information that travels with the packet in its header. Conversely, the packet header is not fully utilised, although it contains much more information than required to simply select the next hop.

Second, packets are typically forwarded along a shortest path route to the destination, usually discovered by routing protocols that use an additive link metric (hop count); the consequence is that paths to a common egress router form a tree rooted at the destination. With heavy loads, links on a shortest path tree may become congested while others remain under-utilised due to uneven traffic distribution. This approach does not exploit alternate paths either to reroute traffic around congested nodes, to perform load balancing or to support dynamic fall over to backup paths. Although it would be possible to redirect traffic by changing link metrics (based for example on traffic characteristics and capacity constraints), this may lead to undesirable effects, such as changing the path of all packets that traverse congested links. In other words, traffic engineering has not been exploited in IP networks [6][7]. Moreover, destination-based forwarding, as used today in IP networks, seriously limits the network services that can be offered, since it does not allow provisioning paths specific to particular sources or services or with a QoS constraint.

Finally, processing the IP header, which includes looking-up the routing table, decrementing the Time To Live (TTL) value and computing a new Cyclic Redundancy Check (CRC) is computationally more intensive and takes more time than processing a label and using it as an index into the forwarding table. Nevertheless, state of the art gigabit routers use optimised algorithms to speed up the IP header processing in hardware.

On the other hand, connection-oriented packet switching is based on the set up of virtual channel connections (virtual circuits) by means of signaling procedures, before data transfer takes place. Packets on a virtual circuit are forwarded along a fixed path, defined at set up; the virtual circuit is identified by a short label in the packets headers. In ATM, the label is structured in two parts and carries a Virtual Path / Virtual Channel Identifier (VPI/VCI), while in Frame Relay the label is a Data Link Connection Identifier (DLCI).

3. MPLS operation

MPLS overcomes the problems found in conventional IP networks as well as the limitations of overlay models, by combining traditional IP routing with a label swapping technique, implemented by separate control and forwarding components. Control functions are responsible for building and maintaining routing tables; routing information is used to create the forwarding tables that define how packets are switched and labels are swapped at each node. Figure 1 is a simplified representation of this model.

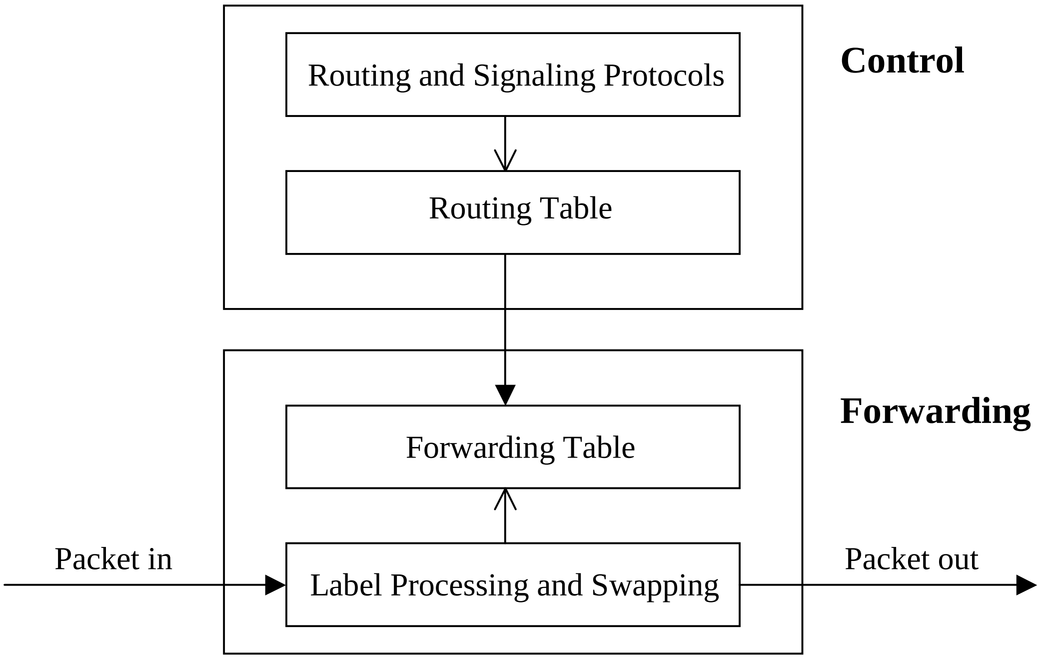

Figure 1 – Mapping between routing and forwarding tables

Forward Equivalence Class

A MPLS label is a short, fixed length identifier, with local significance, used to identify a Forwarding Equivalence Class (FEC). A FEC simply represents a group of IP packets that are forwarded in the same manner, that is, over the same path and receiving the same forwarding treatment, since they have similar transport requirements.

The assignment of a packet to a FEC is done just once, at the ingress node of a MPLS domain, where the packet is classified. It is possible to use a variety of forwarding criteria to assign packets to FECs, besides the conventional address prefix used in destination based routing, such as: Classes of Service (based for instance in fields of IPv4 or IPv6 packet headers), Application flows (requiring both source and destination addresses as well as other Network or Transport layer information), IP multicast groups, explicit routing and VPNs.

A FEC is encoded with a label, which travels with the packet that becomes a "labelled packet". The label may reside in an encapsulation header added for this purpose or may be carried on a layer two header that natively supports labels, such as in ATM or Frame Relay.

Label Switched Paths and Label Switching Routers

Packets in a particular FEC follow a common path through one or more MPLS nodes. This is called a Label Switched Path (LSP), which is defined by the set of labels associated to a FEC at each hop. The forwarding decision taken by each node is simply based on the incoming label, which is used as an index into a table that specifies the next hop and the outgoing label, which is inserted in place of the incoming label (label swapping). A LSP must be set up and labels assigned at each hop before traffic can be forwarded.

This is similar to conventional virtual circuit switching, but differs in the way the forwarding tables are created and maintained. We may also say, by analogy, that in hop-by-hop routing a

FEC represents a common destination address prefix for a group of packets, but each node makes an independent FEC assignment.

By using LSPs, MPLS can provide many of the advantages of connection-oriented networks, while retaining the simplicity of datagram networks.

MPLS nodes are called Label Switching Routers (LSR), but it is usual to refer to the edge (ingress or egress) nodes as Label Edge Routers (LER). All nodes are aware of MPLS control protocols, run the same layer three routing protocols and forward packets based on label processing and swapping at wire speed. In addition, a LER performs some specific and more processing intensive functions, such as interfacing external networks and, in the case of the ingress LER, classifying packets into FECs, assigning the corresponding labels and adding them to the original packet. Therefore, in MPLS, a single family of devices runs the same set of protocols over a common physical and logical topology shared by all nodes, unlike architectures based on the overlay model.

Figure 2 shows a LSP and the basic operations performed on a packet. The ingress LER adds a label to the packet, while the egress LER removes it; labels are assigned independently on each hop and are swapped as the packet moves along the LSP.

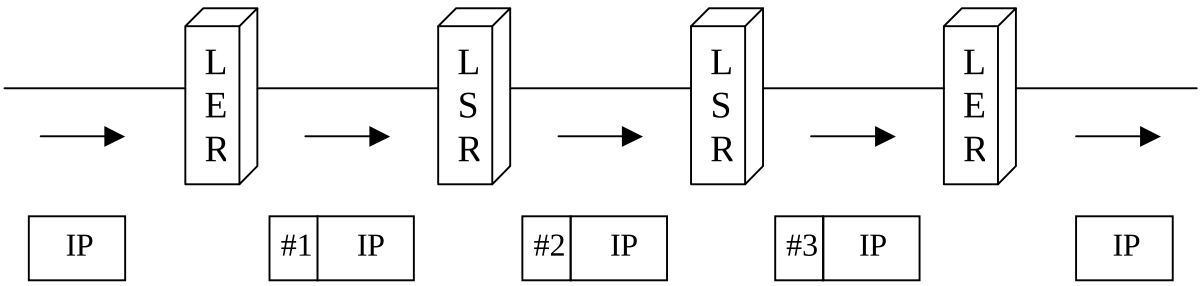

Figure 2 – Label insertion, swapping and removal

Label assignment and distribution

A key issue in MPLS is the binding between labels (that represent a FEC) and a route. Three steps are required: allocating a label, binding it to a route (LSP) and distributing label binding information among LSRs.

IETF has standardized a new signaling protocol, called Label Distribution Protocol (LDP) [12], for set up and maintenance of LSPs. It allows the distribution of label binding information between LSRs, thus ensuring that adjacent nodes share a common FEC to label binding and allowing the creation of LSPs. The forwarding table (label information base) is constructed as the result of label distribution.

Label assignment and route selection are therefore required to set up a LSP. But when should labels be assigned and bound to routes (FECs) and how are routes discovered or selected?

The need for label creation and binding may be driven by data or control traffic. In the data-driven strategy, the arrival of data at a node triggers the process. In the control-driven approach two methods may be adopted, depending on the route selection scheme: labels are assigned once routes are discovered by conventional routing protocols (topology-based) or in response to request-based control traffic, such as Reservation Protocol (RSVP) messages [13]. Control-driven methods are preferred due to their better scalability properties.

Label stack and hierarchical routing

In the previous description, it was considered that an IP packet is encapsulated with a single label. However, MPLS supports a more general mechanism, in which a labelled packet can carry a number of labels organised in a last-in, first-out manner. This is called a label stack.

The use of a label stack allows hierarchical routing, with different levels of granularity, possibly across various domains. Label swapping is always based on the current top level of the stack and adding or removing a label corresponds to normal push and pop operations on the stack. It is also possible to create LSP tunnels that can nest to any depth; one possible application is for the provision of MPLS-based VPNs.

Label encoding

MPLS does not rely upon a specific layer two technology. The only requirement is that LSRs are able to exchange labelled packets across data links.

The IETF defined a general encoding technique that must be supported by LSRs to produce labelled packet before they are transmitted on a data link [14]. It assumes a label stack and is particularly targeted at data links that do not support labels, such as PPP (Point to Point Protocol) or LAN media, although it may be used in Frame Relay or ATM-based LSRs.

A label stack is represented as a sequence of label stack entries. An entry is four octets long and includes four fields:

- Label: Label value (20 bits)
- Exp: Experimental use (3 bits)
- S: Bottom of Stack (1 bit)
- TTL: Time To Live (8 bits)

A possible use of the Exp field is as a Class of Service (CoS) identifier.

In PPP and LAN data links, the label stack entries are inserted between layer two and layer three headers, as illustrated in Figure 3, and constitute what is usually called a shim header.

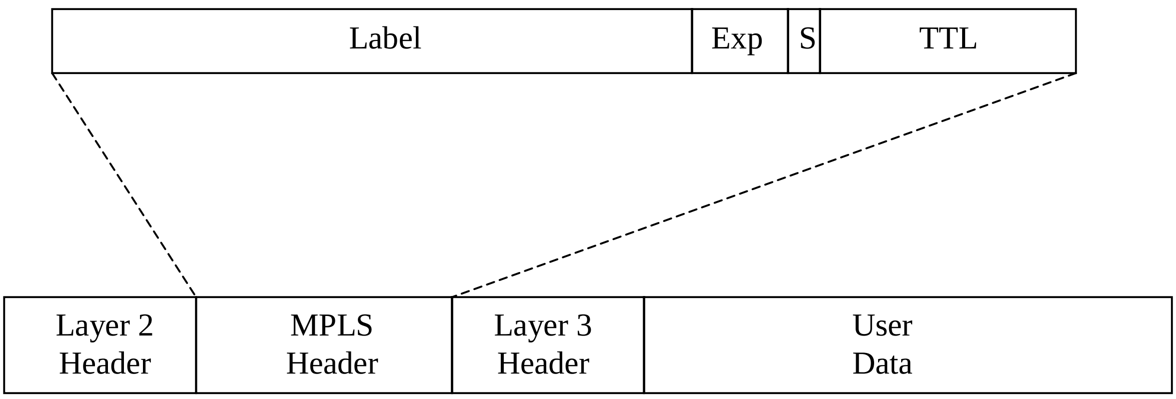

Figure 3 – MPLS header and encapsulation

In ATM-based LSRs, the top label is directly encoded into the VCI and/or VPI field [15]. In general, when a label stack has to be carried, the label stack entries for a particular packet are carried as a shim header in the ATM Adaptation Layer (AAL 5) frame; the actual value of the top label is encoded in the VPI/VCI field of the ATM cells and the label value of the top entry in the shim is set to zero.

Route selection

There are two alternatives to select a route used to set up the LSP for a particular FEC.

The first one is based on hop-by-hop routing; each LSR independently determines the next hop for the LSP based on its IP forwarding table, which is built by traditional IP routing protocols. This is the default, topology-based method and allows the discovery of shortest path routes; a hop-by-hop LSP follows the path that a packet using conventional routing would have used.

The second one is based on explicit routing, which is similar to source routing. In an explicitly routed LSP (ER-LSP), the route for the path is explicitly defined by a single LSR (usually the ingress or egress node) and may include all or only a subset of the LSRs in the path (strict or loose LSP). The route (sequence of LSRs) may be selected by configuration or dynamically and is conveyed in a control message that traverses all nodes along the specified route.

Figure 4 shows an example of a shortest path tree built by hop-by-hop routing and a single ER-LSP.

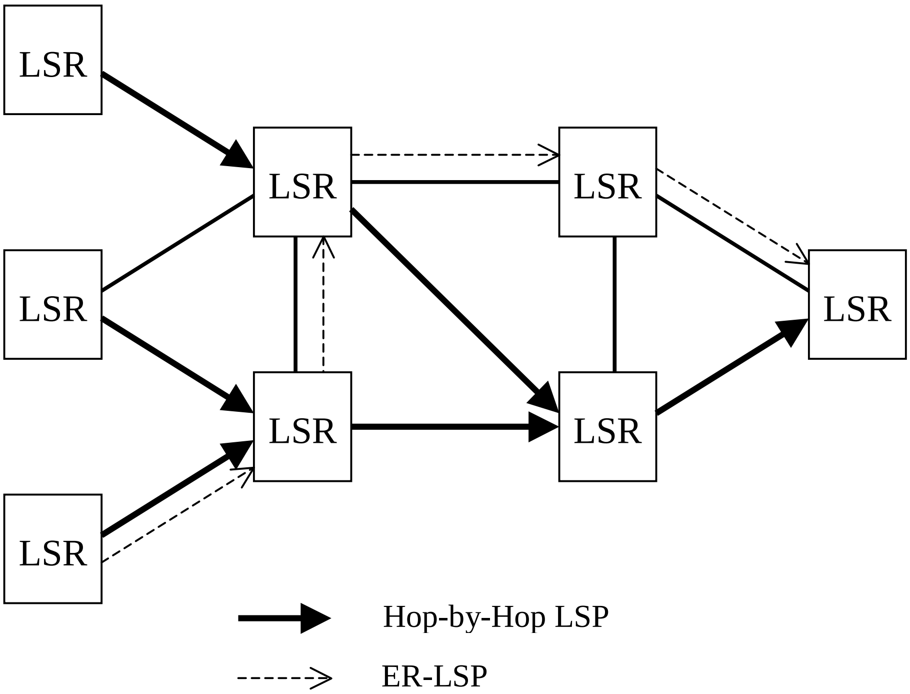

Figure 4 – Hop-by-hop and Explicitly Routed LSPs

4. Traffic Engineering and QoS in MPLS Networks

An ER-LSP may be defined and controlled by the network operator or a network management application. Based on administrative or QoS policies or traffic engineering requirements, some traffic can be forced into a path different from the shortest path computed by a routing protocol.

Two approaches have been considered to control (establish, terminate, re-route) ER-LSPs: extending the capabilities of the Label Distribution Protocol to include explicit paths via Constraint-based Routed Label Distribution Protocol (CR-LDP) [16] and extensions to RSVP (MPLS-RSVP) [17]. At an abstract level the functions of CR-LDP and MPLS-RSVP are rather similar. They both allow a LSR to trigger and control the establishment of a LSP between itself and a remote LER, to strictly or loosely specify the route to be taken by the LSP, and to specify queuing and scheduling parameters to be associated with this LSP at every hop. The relative advantages and disadvantages of these two schemes can be found in [18], but manageable traffic engineering and QoS control cannot be realized unless one of these protocols is deployed.

By allowing the network to explicitly route a LSP, both traffic engineering techniques [6][7] and provisioning of differentiated services can be supported in a MPLS domain.

Constraint-based routing and QoS

Explicit routing is a particular case of constraint-based routing, where the constraint is the explicit path. In general, constraint-based routing may take into account link characteristics, such as bandwidth or delay, hop count or QoS parameters.

MPLS allows setting up explicit paths and forwarding traffic on them, but it does not provide the means to find out paths with constraints. Because MPLS allows traffic engineering and explicit routing, there is keen interest in QoS routing that allows selection of routes subject to QoS requirements and other policies, instead of the least cost or shortest path route found by traditional routing protocols.

In order to allow the computation of routes with constraints, it is necessary to extend the Interior Gateway Protocols to carry additional information about links that can include, for example, maximum link bandwidth, maximum reservable bandwidth, current bandwidth reservation at different priority levels, default traffic engineering metric or other attributes used for policy-based routing. Constraint-based routing is just one of the issues being considered by the IETF in the context of Internet Traffic Engineering [19].

MPLS and Differentiated Services

In a DiffServ domain all packets requiring the same behaviour constitute a Behaviour Aggregate (BA). At the ingress node of a DiffServ domain packets are classified and marked with a DiffServ Code Point (DSCP) corresponding to their BA. The DSCP is used in each node visited by the packet to select the Per Hop Behaviour (PHB) that determines the treatment it will receive (scheduling, drop precedence, etc.).

MPLS support of Differentiated Services has already been addressed by the IETF [20] and several alternatives solutions were considered for mapping BAs onto LSPs. The simplest one consists in using a single LSP to support up to eight BAs of a given FEC. In this case the Exp field of the MPLS shim header is used by each LSR to determine the PHB to be applied to the packet.

One promising solution for QoS provisioning in IP networks, using currently available IETF standards, consists in combining MPLS with differentiated services and constraint-based routing.

5. Conclusions

It is commonly accepted that MPLS offers many advantages over earlier network solutions, as a carrier infrastructure capable of service integration and differentiation. However, some of these advantages are not exclusive of MPLS; on the other hand, MPLS must be combined with other mechanisms, such as QoS, to make use of its features.

Therefore, when evaluating the merits of MPLS, it is useful to adopt a critical view [21], especially when considering competitive solutions, such as gigabit routers, which also promise high throughput and fast switching, as well as traffic and QoS differentiation.

Nevertheless, there are some strong arguments in favour of MPLS. In general, MPLS offers a unique combination of some attractive properties:

- scalability, in terms of number of nodes and traffic flows;
- flexibility, since it is not tied to a single forwarding technology;
- simple and fast label forwarding, which improves network performance;
- capability to support traffic engineered paths and service differentiation, essential for QoS provisioning.

In particular, properties like scalability and traffic engineering are especially valuable, when considering short-term deployment of MPLS. This allows offering an efficient transit core network (with high throughput and low latency), improved economy of scale, new services (such as CoS-based forwarding and VPNs), and the ability for fast restoration of data traffic.

Another important issue, in view of the investment many carriers have recently done in ATM equipment, is the possibility of leveraging the installed ATM infrastructure – either using ATM-based LSRs or running MPLS over ATM (overlay model), in which case MPLS LSRs communicate over an ATM cloud.

Although MPLS has been targeted at WAN environments, it can be used in LANs, as well. In this case it is an alternative to solutions based on conventional layer two LAN switches and multilayer switches (or router switches) or to ATM LANs based on overlay models, such as LAN Emulation (LANE) and Multiprotocol over ATM (MPOA).

When considering the deployment of MPLS in the Internet, it is necessary to take into account the profound consequences at the architectural level. It changes the basic forwarding model, which has remained essentially unaltered since the early days of the ARPANET. It also impacts the routing architecture, requiring that routing protocols perform new and more complex tasks.

Short-term applications are likely to be within a single network administrative domain; over time, interdomain MPLS is likely to occur, with transit carriers providing services to local or national Internet Service Providers.

The long-term evolution is yet unclear. In order to fully exploit the benefits of MPLS, some open issues still need to be answered and are object of intense research. In particular, it is necessary to specify, test and validate, in operational conditions, criteria and mechanisms for selecting routes and dynamically establishing paths according to traffic engineering or QoS policies, as well as managing their QoS characteristics.

References

[1] Xipeng Xiao *et al*., "Internet QoS: a Big Picture", IEEE Network Magazine, Vol. 13, No. 2, March/April 1999, pp. 8-18.

[2] R. Braden *et al*., "Integrated Services in the Internet Architecture: an Overview", RFC 1633, June 1994.

[3] S. Blake *et al*., "An Architecture for Differentiated Services", RFC 2475, December 1998.

[4] ATM Forum, "Traffic Management Specification", Version 4.1, AF-TM-0121.000, March 1999.

[5] M. Laubach *et al*., "Classical IP and ARP over ATM", RFC 2225, April 1998.

[6] Daniel O. Awduche, "MPLS and Traffic Engineering in IP Networks", IEEE Communications Magazine, Vol. 37, No. 12, December 1999, pp. 42-47.

[7] George Swallow, "MPLS Advantages for Traffic Engineering", IEEE Communications Magazine, Vol. 37, No. 12, December 1999, pp. 54-57.

[8] Peter Newman *et al*., "IP Switching and Gigabit Routers", IEEE Communications Magazine, Vol. 35, No. 1, January 1997, pp. 64-69.

[9] Yakov Rekhter *et al*., "Tag Switching Architecture Overview", Proceedings of the IEEE, Vol. 85, No. 12, December 1997, pp. 1973-1983.

[10] E. Rosen *et al*., "Multiprotocol Label Switching Architecture", RFC 3031, January 2001.

[11] D. Awduche *et al*., "Requirements for Traffic Engineering over MPLS", RFC 2702, September 1999.

[12] L. Andersson *et al*., "LDP Specification", RFC 3036, January 2001.

[13] R. Braden *et al*., "Resource ReSerVation Protocol (RSVP) - Version 1 Functional Specification", RFC 2205, September 1997.

[14] E. Rosen *et al*., "MPLS Label Stack Encoding", RFC 3032, January 2001.

[15] B. Davie *et al*., "MPLS using LDP and ATM VC Switching", RFC 3035, January 2001.

[16] B. Jamoussi *et al*., "Constraint-Based LSP Setup using LDP", RFC 3212, January 2002.

[17] D. Awduche *et al*., “RSVP-TE: Extensions to RSVP for LSP Tunnels”, RFC 3209, December 2001.

[18] Anoop Ghanwani *et al*., “Traffic Engineering Standards in IP Networks using MPLS”, IEEE Communications Magazine, Vol. 37, No. 12, December 1999, pp. 49-53.

[19] D. Awduche *et al*., “Overview and principles of Internet Traffic Engineering”, RFC 3272, May 2002.

[20] F. Le Faucheur *et al*., “MPLS Support of Differentiated Services”, RFC 3270, May 2002.

[21] Grenville Armitage, “MPLS: The Magic Behind the Myths”, IEEE Communications Magazine, Vol. 38, No. 1, January 2000, pp. 124-131.