CHAPTER 1

Multiprotocol Label Switching (MPLS) Architecture Overview

Traditional IP packet forwarding analyzes the destination IP address contained in the network layer header of each packet as the packet travels from its source to its final destination. A router analyzes the destination IP address independently at each hop in the network. Dynamic routing protocols or static configuration builds the database needed to analyze the destination IP address (the routing table). The process of implementing traditional IP routing also is called *hop-by-hop destination-based unicast routing*.

Although successful, and obviously widely deployed, certain restrictions, which have been realized for some time, exist for this method of packet forwarding that diminish its flexibility. New techniques are therefore required to address and expand the functionality of an IP-based network infrastructure.

This first chapter concentrates on identifying these restrictions and presents a new architecture, known as *Multiprotocol Label Switching (MPLS)*, that provides solutions to some of these restrictions. The following chapters focus first on the details of the MPLS architecture in a pure router environment, and then in a mixed router/ATM switch environment.

Scalability and Flexibility of IP-based Forwarding

To understand all the issues that affect the scalability and the flexibility of traditional IP packet forwarding networks, you must start with a review of some of the basic IP forwarding mechanisms and their interaction with the underlying infrastructure (local- or wide-area networks). With this information, you can identify any drawbacks to the existing approach and perhaps provide alternative ideas on how this could be improved.

Network Layer Routing Paradigm

Traditional network layer packet forwarding (for example, forwarding of IP packets across the Internet) relies on the information provided by network layer routing protocols (for example, Open Shortest Path First [OSPF] or Border Gateway Protocol [BGP]), or static routing, to make an independent forwarding decision at each hop (router) within the network. The forwarding decision is based solely on the destination unicast IP address. All packets for the same destination follow the same path across the network if no other equal-cost paths exist. Whenever a router has two equal-cost paths toward a destination, the packets toward the destination might take one or both of them, resulting in some degree of load sharing.

NOTE Enhanced Interior Gateway Routing Protocol (EIGRP) also supports non–equal-cost load sharing although the default behavior of this protocol is equal-cost. You must configure EIGRP *variance* for non–equal-cost load balancing. Please see *EIGRP Network Design Solutions* (ISBN 1-57870-165-1), from Cisco Press for more details on EIGRP.

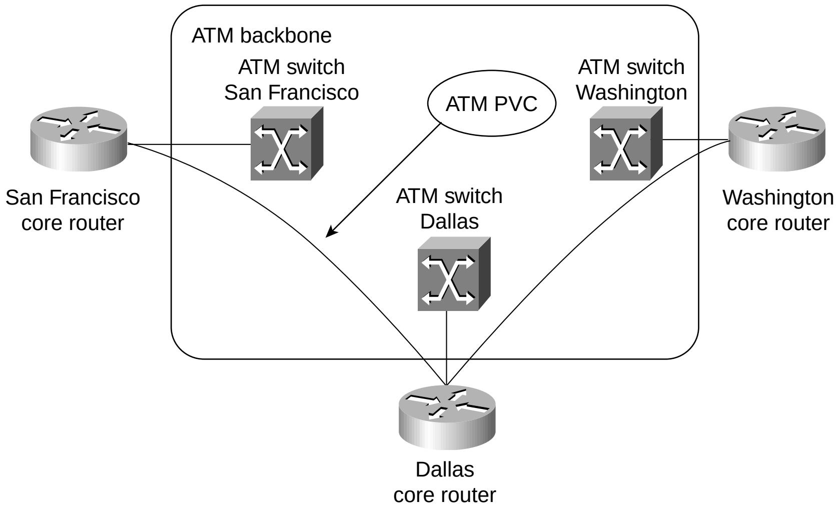
Load sharing in Cisco IOS can be performed on a packet-by-packet or source-destination-pair basis (with Cisco Express Forwarding [CEF] switching) or on a destination basis (most of the other switching methods).

Routers perform the decision process that selects what path a packet takes. These network layer devices participate in the collection and distribution of network-layer information, and perform Layer 3 switching based on the contents of the network layer header of each packet. You can connect the routers directly by point-to-point links or local-area networks (for example, shared hub or MAU), or you can connect them by LAN or WAN switches (for example, Frame Relay or ATM switches). These Layer 2 (LAN or WAN) switches unfortunately do not have the capability to hold Layer 3 routing information or to select the path taken by a packet through analysis of its Layer 3 destination address. Thus, Layer 2 (LAN or WAN) switches cannot be involved in the Layer 3 packet forwarding decision process. In the case of the WAN environment, the network designer has to establish Layer 2 paths manually across the WAN network. These paths then forward Layer 3 packets between the routers that are connected physically to the Layer 2 network.

LAN Layer 2 paths are simple to establish—all LAN switches are transparent to the devices connected to them. The WAN Layer 2 path establishment is more complex. WAN Layer 2 paths usually are based on a point-to-point paradigm (for example, virtual circuits in most WAN networks) and are established only on request through manual configuration. Any routing device (ingress router) at the edge of the Layer 2 network that wants to forward Layer 3 packets to any other routing device (egress router) therefore needs to either establish a direct connection across the network to the egress device or send its data to a different device for transmission to the final destination.

Consider, for example, the network shown in Figure 1-1.

Figure 1-1 *Sample IP Network Based on ATM Core*



The network illustrated in Figure 1-1 is based on an ATM core surrounded by routers that perform network layer forwarding. Assuming that the only connections between the routers are the ones shown in Figure 1-1, all the packets sent from San Francisco to or via Washington must be sent to the Dallas router, where they are analyzed and sent back over the same ATM connection in Dallas to the Washington router. This extra step introduces delay in the network and unnecessarily loads the CPU of the Dallas router as well as the ATM link between the Dallas router and the adjacent ATM switch in Dallas.

To ensure optimal packet forwarding in the network, an ATM virtual circuit must exist between any two routers connected to the ATM core. Although this might be easy to achieve in small networks, such as the one in Figure 1-1, you run into serious scalability problems in large networks where several tens or even hundreds of routers connect to the same WAN core.

The following facts illustrate the scalability problems you might encounter:

- Every time a new router is connected to the WAN core of the network, a virtual circuit must be established between this router and any other router, if optimal routing is required.

Note In Frame Relay networks, the entire configuration could be done within the Layer 2 WAN core and the routers would find new neighbors and their Layer 3 protocol addresses through the use of LMI and Inverse ARP. This also is possible on an ATM network through the use of Inverse ARP, which is enabled by default when a new PVC is added to the configuration of the router, and ILMI, which can discover PVCs dynamically that are configured on the local ATM switch.

- With certain routing protocol configurations, every router attached to the Layer 2 WAN core (built with ATM or Frame Relay switches) needs a dedicated virtual circuit to every other router attached to the same core. To achieve the desired core redundancy, every router also must establish a routing protocol adjacency with every other router attached to the same core. The resulting full-mesh of router adjacencies results in every router having a large number of routing protocol neighbors, resulting in large amounts of routing traffic. For example, if the network runs OSPF or IS-IS as its routing protocol, every router propagates every change in the network topology to every other router connected to the same WAN backbone, resulting in routing traffic proportional to the *square* of the number of routers.

Note Configuration tools exist in recent Cisco IOS implementations of IS-IS and OSPF routing protocols that allow you to reduce the routing protocol traffic in the network. Discussing the design and the configuration of these tools is beyond the scope of this book (any interested reader should refer to the relevant Cisco IOS configuration guides).

- Provisioning of the virtual circuits between the routers is complex, because it’s very hard to predict the exact amount of traffic between any two routers in the network. To simplify the provisioning, some service providers just opt for lack of service guarantee in the network—zero Committed Information Rate (CIR) in a Frame Relay network or Unspecified Bit Rate (UBR) connections in an ATM network.

The lack of information exchange between the routers and the WAN switches was not an issue for traditional Internet service providers that used router-only backbones or for traditional service providers that provided just the WAN services (ATM or Frame Relay virtual circuits). There are, however, several drivers that push both groups toward mixed backbone designs:

- Traditional service providers are asked to offer IP services. They want to leverage their investments and base these new services on their existing WAN infrastructure.
- Internet service providers are asked to provide tighter quality of service (QoS) guarantees that are easier to meet with ATM switches than with traditional routers.

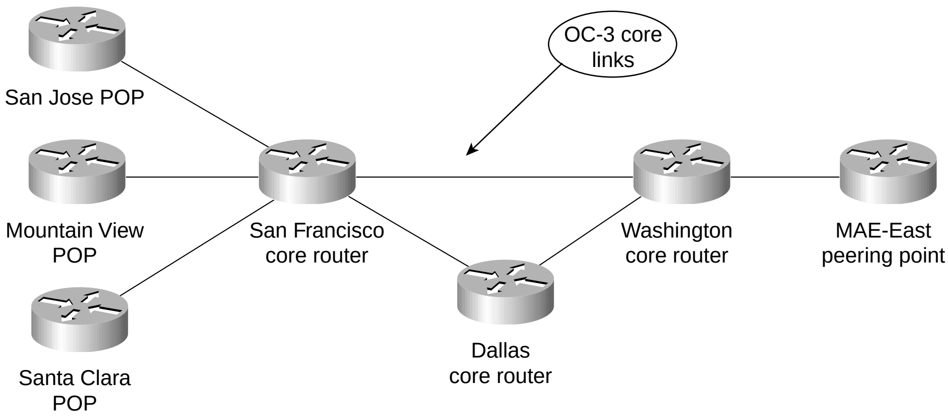
- The rapid increase in bandwidth requirements prior to the introduction of optical router interfaces forced some large service providers to start relying on ATM technology because the router interfaces at that time did not provide the speeds offered by the ATM switches.

It is clear, therefore, that a different mechanism must be used to enable the exchange of network layer information between the routers and the WAN switches and to allow the switches to participate in the decision process of forwarding packets so that direct connections between edge routers are no longer required.

Differentiated Packet Servicing

Conventional IP packet forwarding uses only the IP destination address contained within the Layer 3 header within a packet to make a forwarding decision. The hop-by-hop destination-only paradigm used today prevents a number of innovative approaches to network design and traffic-flow optimization. In Figure 1-2, for example, the direct link between the San Francisco core router and the Washington core router forwards the traffic entering the network in any of the Bay Area Points-of-Presence (POPs), although that link might be congested and the links from San Francisco to Dallas and from Dallas to Washington might be only lightly loaded.

Figure 1-2 *Sample Network that Would Benefit from Traffic Engineering*



Although certain techniques exist to affect the decision process, such as Policy Based Routing (PBR), no single scalable technique exists to decide on the full path a packet takes across the network to its final destination. In the network shown in Figure 1-2, the policy-based routing must be deployed on the San Francisco core router to divert some of the Bay Area to Washington traffic toward Dallas. Deploying such features as PBR on core routers could severely reduce the performance of a core router and result in a rather unscalable network design. Ideally, the edge routers (for example, the Santa Clara POP in Figure 1-2) can specify over which core links the packets should flow.

NOTE Several additional issues are associated with policy-based routing. PBR can lead easily to forwarding loops as a router configured with PBR deviates from the forwarding path learned from the routing protocols. PBR also is hard to deploy in large networks; if you configure PBR at the edge, you must be sure that *all* routers in the forwarding path can make the *same* route selection.

Because most major service providers deploy networks with redundant paths, a requirement clearly exists to allow the ingress routing device to be capable of deciding on packet forwarding, which affects the path a packet takes across the network, and of applying a *label* to that packet that indicates to other devices which path the packet should take.

This requirement also should allow packets that are destined for the same IP network to take separate paths instead of the path determined by the Layer 3 routing protocol. This decision also should be based on factors other than the destination IP address of the packet, such as from which port the packet was learned, what quality of service level the packet requires, and so on.

Independent Forwarding and Control

With conventional IP packet forwarding, any change in the information that controls the forwarding of packets is communicated to all devices within the routing domain. This change always involves a period of convergence within the forwarding algorithm.

A mechanism that can change how a packet is forwarded, without affecting other devices within the network, certainly is desirable. To implement such a mechanism, forwarding devices (routers) should not rely on IP header information to forward the packet; thus, an additional label must be attached to a forwarded packet to indicate its desired forwarding behavior. With the packet forwarding being performed based on labels attached to the original IP packets, any change within the decision process can be communicated to other devices through the distribution of new labels. Because these devices merely forward traffic based on the attached label, a change should be able to occur without any impact at all on any devices that perform packet forwarding.

External Routing Information Propagation

Conventional packet forwarding within the core of an IP network requires that external routing information be advertised to all transit routing devices. This is necessary so that packets can be routed based on the destination address that is contained within the network layer header of the packet. To continue the example from previous sections, the core routers in Figure 1-2 would have to store all Internet routes so that they could propagate packets between Bay Area customers and a peering point in MAE-East.

NOTE You might argue that each major service provider also must have a peering point somewhere on the West coast. That fact, although true, is not relevant to this discussion because you can always find a scenario where a core router with no customers or peering partners connected to it needs complete routing information to be able to forward IP packets correctly.

This method has scalability implications in terms of route propagation, memory usage, and CPU utilization on the core routers, and is not really a required function if all you want to do is pass a packet from one edge of the network to another.

A mechanism that allows internal routing devices to *switch* the packets across the network from an ingress router toward an egress router without analyzing network layer destination addresses is an obvious requirement.

Multiprotocol Label Switching (MPLS) Introduction

Multiprotocol Label Switching (MPLS) is an emerging technology that aims to address many of the existing issues associated with packet forwarding in today's Internetworking environment. Members of the IETF community worked extensively to bring a set of standards to market and to evolve the ideas of several vendors and individuals in the area of *label switching*. The IETF document *draft-ietf-mpls-framework* contains the framework of this initiative and describes the primary goal as follows:

> The primary goal of the MPLS working group is to standardize a base technology that integrates the label swapping forwarding paradigm with network layer routing. This base technology (label swapping) is expected to improve the price/performance of network layer routing, improve the scalability of the network layer, and provide greater flexibility in the delivery of (new) routing services (by allowing new routing services to be added without a change to the forwarding paradigm).

NOTE You can download IETF working documents from the IETF home page (www.ietf.org). For MPLS working documents, start at the MPLS home page (www.ietf.org/html.charters/mpls-charter.html).

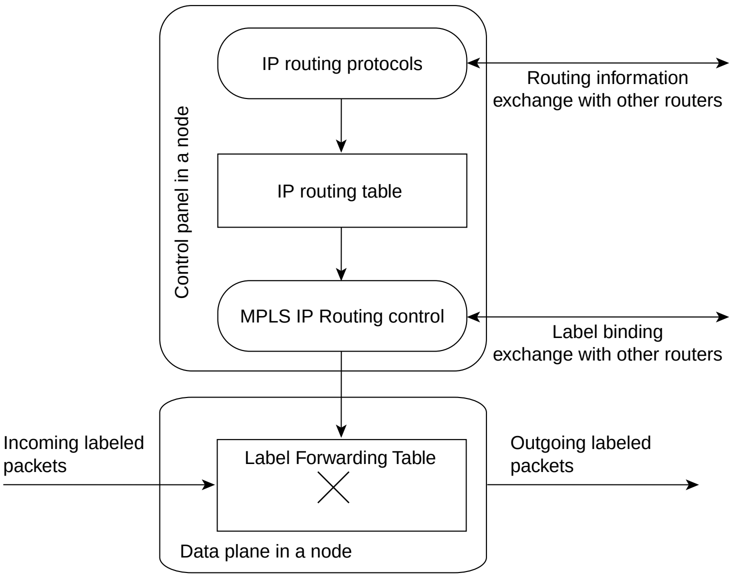
The MPLS architecture describes the mechanisms to perform label switching, which combines the benefits of packet forwarding based on Layer 2 switching with the benefits of Layer 3 routing. Similar to Layer 2 networks (for example, Frame Relay or ATM), MPLS assigns *labels* to packets for transport across packet- or cell-based networks. The forwarding mechanism throughout the network is *label swapping*, in which units of data (for example, a packet or a cell) carry a short, fixed-length label that tells switching nodes along the packets path how to process and forward the data.

The significant difference between MPLS and traditional WAN technologies is the way labels are assigned and the capability to carry a stack of labels attached to a packet. The concept of a label stack enables new applications, such as Traffic Engineering, Virtual Private Networks, fast rerouting around link and node failures, and so on.

Packet forwarding in MPLS is in stark contrast to today's connectionless network environment, where each packet is analyzed on a hop-by-hop basis, its layer 3 header is checked, and an independent forwarding decision is made based on the information extracted from a network layer routing algorithm.

The architecture is split into two separate components: the *forwarding* component (also called the *data plane*) and the control component (also called the *control plane*). The forwarding component uses a label-forwarding database maintained by a label switch to perform the forwarding of data packets based on labels carried by packets. The control component is responsible for creating and maintaining label-forwarding information (referred to as *bindings*) among a group of interconnected label switches. Figure 1-3 shows the basic architecture of an MPLS node performing IP routing.

Figure 1-3 *Basic Architecture of an MPLS Node Performing IP Routing*



Every MPLS node must run one or more IP routing protocols (or rely on static routing) to exchange IP routing information with other MPLS nodes in the network. In this sense, every MPLS node (including ATM switches) is an IP router on the control plane.

Similar to traditional routers, the IP routing protocols populate the IP routing table. In traditional IP routers, the IP routing table is used to build the IP forwarding cache (fast switching cache in Cisco IOS) or the IP forwarding table (Forwarding Information Base [FIB] in Cisco IOS) used by Cisco Express Forwarding (CEF).

In an MPLS node, the IP routing table is used to determine the label binding exchange, where adjacent MPLS nodes exchange labels for individual subnets that are contained within the IP routing table. The label binding exchange for unicast destination-based IP routing is performed using the Cisco proprietary Tag Distribution Protocol (TDP) or the IETF-specified Label Distribution Protocol (LDP).

The MPLS IP Routing Control process uses labels exchanged with adjacent MPLS nodes to build the Label Forwarding Table, which is the forwarding plane database that is used to forward labeled packets through the MPLS network.

MPLS Architecture—The Building Blocks

As with any new technology, several new terms are introduced to describe the devices that make up the architecture. These new terms describe the functionality of each device and their roles within the MPLS domain structure.

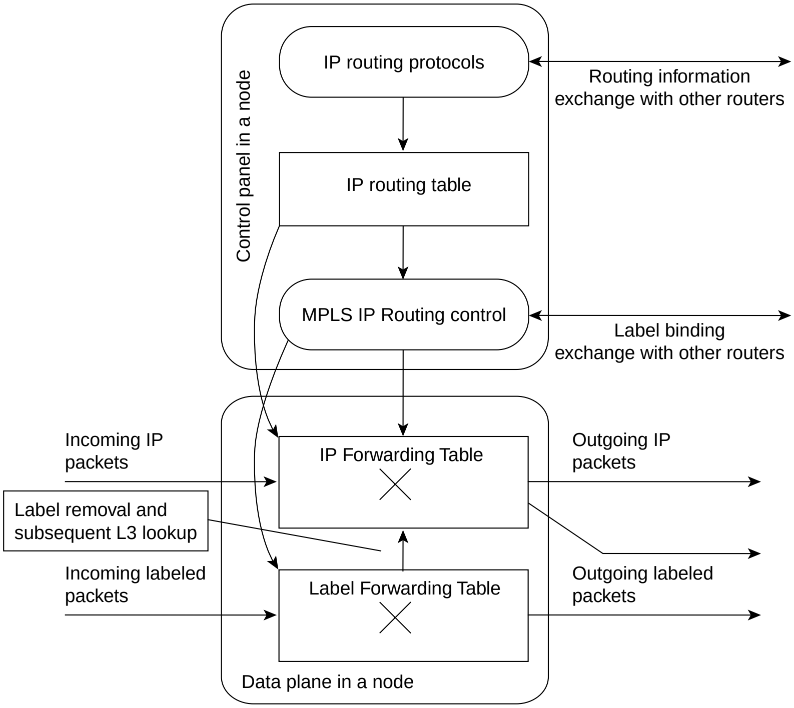
The first device to be introduced is the *Label Switch Router (LSR)*. Any router or switch that implements label distribution procedures and can forward packets based on labels falls under this category. The basic function of label distribution procedures is to allow an LSR to distribute its label bindings to other LSRs within the MPLS network. (Chapter 2, “Frame-mode MPLS Operation,” discusses label distribution procedures in detail.)

Several different types of LSR exist that are differentiated by what functionality they provide within the network infrastructure. These different types of LSR are described within the architecture as *Edge-LSR*, *ATM-LSR*, and *ATM edge-LSR*. The distinction between various LSR types is purely architectural—a single box can serve several of the roles.

An Edge-LSR is a router that performs either label imposition (sometimes also referred to as *push* action) or label disposition (also called *pop* action) at the edge of the MPLS network. Label imposition is the act of prepending a label, or a stack of labels, to a packet in the ingress point (in respect of the traffic flow from source to destination) of the MPLS domain. Label disposition is the reverse of this and is the act of removing the last label from a packet at the egress point before it is forwarded to a neighbor that is outside the MPLS domain.

Any LSR that has any non-MPLS neighbors is considered an Edge-LSR. However, if that LSR has any interfaces that connect through MPLS to an ATM-LSR, then it also is considered to be an ATM edge-LSR. Edge-LSRs use a traditional IP forwarding table, augmented with labeling information, to label IP packets or to remove labels from labeled packets before sending them to non-MPLS nodes. Figure 1-4 shows the architecture of an Edge-LSR.

Figure 1-4 *Architecture of an Edge-LSR*



An Edge-LSR extends the MPLS node architecture from Figure 1-3 with additional components in the data plane. The standard IP forwarding table is built from the IP routing table and is extended with labeling information. Incoming IP packets can be forwarded as pure IP packets to non-MPLS nodes or can be labeled and sent out as labeled packets to other MPLS nodes. The incoming labeled packets can be forwarded as labeled packets to other MPLS nodes. For labeled packets destined for non-MPLS nodes, the label is removed and a Layer 3 lookup (IP forwarding) is performed to find the non-MPLS destination.

An ATM-LSR is an ATM switch that can act as an LSR. The Cisco Systems, Inc. LS1010 and BPX family of switches are examples of this type of LSR. As you see in the following chapters, the ATM-LSR performs IP routing and label assignment in the control plane and forwards the data packets using traditional ATM cell switching mechanisms on the data plane. In other words, the ATM switching matrix of an ATM switch is used as a Label Forwarding Table of an MPLS node. Traditional ATM switches, therefore, can be redeployed as ATM-LSRs through a software upgrade of their control component.

Table 1-1 summarizes the functions performed by different LSR types. Please note that any individual device in the network can perform more than one function (for example, it can be Edge-LSR and ATM edge-LSR at the same time).

Table 1-1 *Actions Performed by Various LSR Types*

LSR Type	Actions Performed by This LSR Type
LSR	Forwards labeled packets.
Edge-LSR	Can receive an IP packet, perform Layer 3 lookups, and impose a label stack before forwarding the packet into the LSR domain. Can receive a labeled packet, remove labels, perform Layer 3 lookups, and forward the IP packet toward its next-hop.
ATM-LSR	Runs MPLS protocols in the control plane to set up ATM virtual circuits. Forwards labeled packets as ATM cells.
ATM edge-LSR	Can receive a labeled or unlabeled packet, segment it into ATM cells, and forward the cells toward the next-hop ATM-LSR. Can receive ATM cells from an adjacent ATM-LSR, reassemble these cells into the original packet, and then forward the packet as a labeled or unlabeled packet.

Label Imposition at the Network Edge

Label imposition has been described already as the act of prepending a label to a packet as it enters the MPLS domain. This is an edge function, which means that packets are labeled before they are forwarded to the MPLS domain.

To perform this function, an Edge-LSR needs to understand where the packet is headed and which label, or stack of labels, it should assign to the packet. In conventional layer 3 IP forwarding, each hop in the network performs a lookup in the IP forwarding table for the IP destination address contained in the layer 3 header of the packet. It selects a next hop IP address for the packet at each iteration of the lookup and eventually sends the packet out of an interface toward its final destination.

NOTE Some forwarding mechanisms, such as CEF, allow the router to associate each destination prefix known in the routing table to the adjacent next-hop of the destination prefix, thus solving the recursive lookup problem. The whole recursion is resolved while the router populates the cache or the forwarding table and not when it has to forward packets.

Choosing the next hop for the IP packet is a combination of two functions. The first function partitions the entire set of possible packets into a set of IP destination prefixes. The second function maps each IP destination prefix to an IP next hop address. This means that each destination in the network is reachable by one path in respect to traffic flow from one ingress device to the destination egress device (multiple paths might be available if load balancing is performed using equal-cost paths or unequal-cost paths as with some IGP protocols, such as Enhanced IGRP).

Within the MPLS architecture, the results of the first function are known as *Forwarding Equivalence Classes (FECs)*. These can be visualized as describing a group of IP packets that are forwarded in the same manner, over the same path, with the same forwarding treatment.

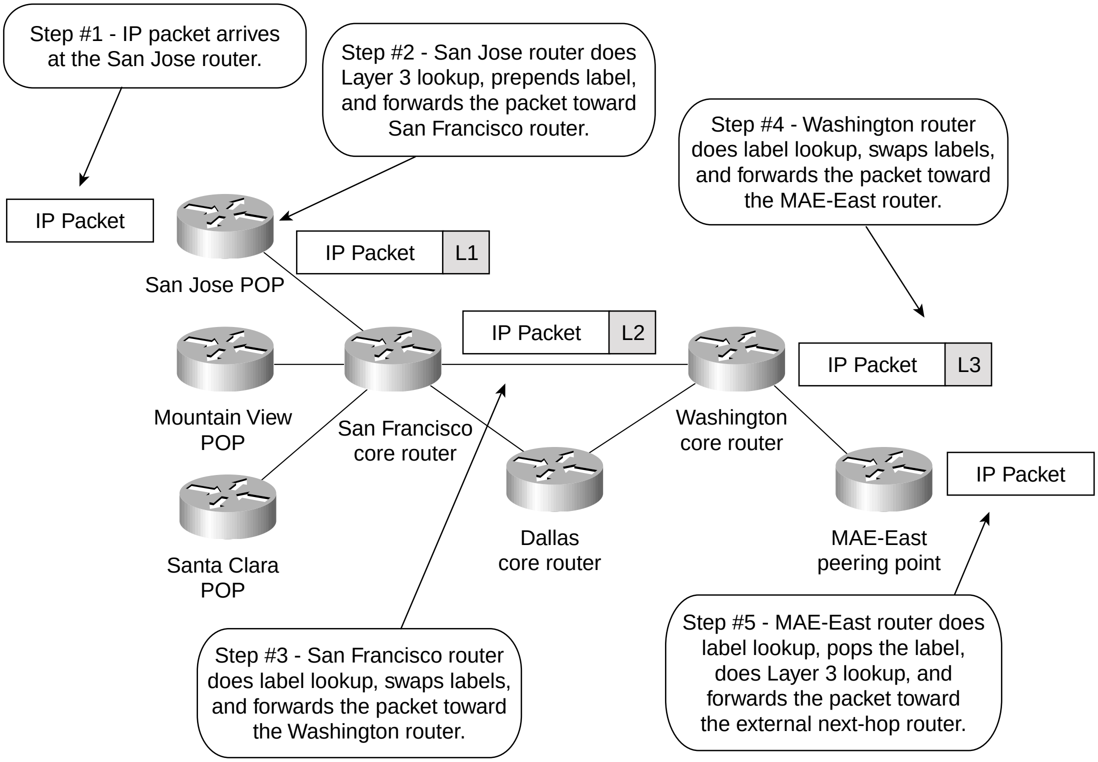
NOTE A Forwarding Equivalence Class might correspond to a destination IP subnet, but also might correspond to any traffic class that the Edge-LSR considers significant. For example, all interactive traffic toward a certain destination or all traffic with a certain value of IP precedence might constitute an FEC. As another example, an FEC can be a subset of the BGP table, including all destination prefixes reachable through the same exit point (egress BGP router).

With conventional IP forwarding, the previously described packet processing is performed at each hop in the network. However, when MPLS is introduced, a particular packet is assigned to a particular FEC just once, and this is at the edge device as the packet enters the network. The FEC to which the packet is assigned is then encoded as a short fixed-length identifier, known as a label.

When a packet is forwarded to its next hop, the label is prepended already to the IP packet so that the next device in the path of the packet can forward it based on the encoded label rather than through the analysis of the Layer 3 header information. Figure 1-5 illustrates the whole process of label imposition and forwarding.

NOTE The actual packet forwarding between the Washington and MAE-East routers might be slightly different from the one shown in Figure 1-5 due to a mechanism called *penultimate hop popping (PHP)*. Penultimate hop popping arguably might improve the switching performance, but does not impact the logic of label switching. Chapter 2 covers this mechanism and its implications.

Figure 1-5 *MPLS Label Imposition and Forwarding*



MPLS Packet Forwarding and Label Switched Paths

Each packet enters an MPLS network at an ingress LSR and exits the MPLS network at an egress LSR. This mechanism creates what is known as an *Label Switched Path (LSP)*, which essentially describes the set of LSRs through which a labeled packet must traverse to reach the egress LSR for a particular FEC. This LSP is unidirectional, which means that a different LSP is used for return traffic from a particular FEC.

The creation of the LSP is a connection-oriented scheme because the path is set up prior to any traffic flow. However, this connection setup is based on topology information rather than a requirement for traffic flow. This means that the path is created regardless of whether any traffic actually is required to flow along the path to a particular set of FECs.

As the packet traverses the MPLS network, each LSR swaps the incoming label with an outgoing label, much like the mechanism used today within ATM where the VPI/VCI is swapped to a different VPI/VCI pair when exiting the ATM switch. This continues until the last LSR, known as the egress LSR, is reached.

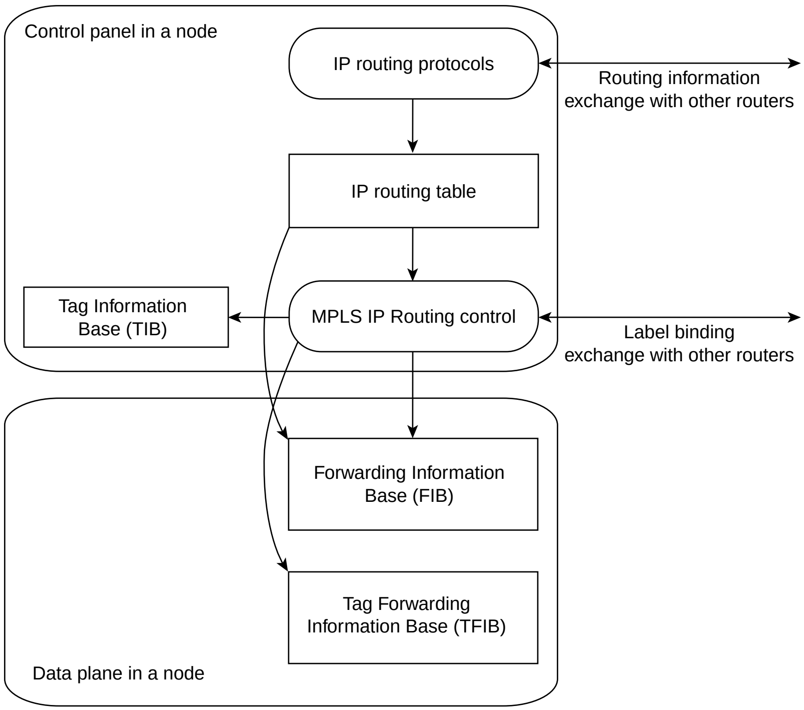
Each LSR keeps two tables, which hold information that is relevant to the MPLS forwarding component. The first, known in Cisco IOS as the *Tag Information Base (TIB)* or *Label Information Base (LIB)* in standard MPLS terms, holds all labels assigned by this LSR and the mappings of these labels to labels received from any neighbors. These label mappings are distributed through the use of label-distribution protocols, which Chapter 2 discusses in more detail.

Just as multiple neighbors can send labels for the same IP prefix but might not be the actual IP next hop currently in use in the routing table for the destination, not all the labels within the TIB/LIB need to be used for packet forwarding. The second table, known in Cisco IOS as the *Tag Forwarding Information Base (TFIB)* or *Label Forwarding Information Base (LFIB)* in MPLS terms, is used during the actual forwarding of packets and holds only labels that are in use currently by the forwarding component of MPLS.

NOTE Label Forwarding Information Base is the MPLS equivalent of the switching matrix of an ATM switch.

Using Cisco IOS terms and Cisco Express Forwarding (CEF) terminology, the Edge-LSR architecture in Figure 1-4 can be redrawn as shown in Figure 1-6 (Edge-LSR was chosen because its function is a superset of non–Edge-LSR).

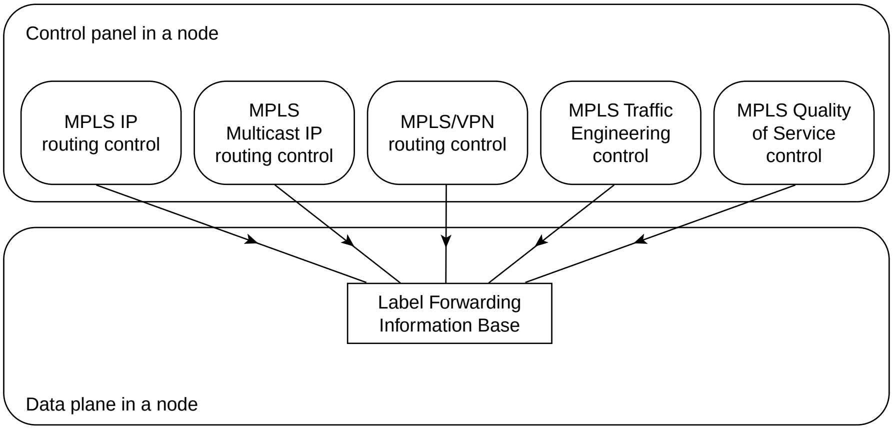
Figure 1-6 *Edge-LSR Architecture Using Cisco IOS Terms*



Other MPLS Applications

The MPLS architecture, as discussed so far, enables the smooth integration of traditional routers and ATM switches in a unified IP backbone (IP+ATM architecture). The real power of MPLS, however, lies in other applications that were made possible, ranging from traffic engineering to peer-to-peer Virtual Private Networks. All MPLS applications use control-plane functionality similar to the IP routing control plane shown in Figure 1-6 to set up the label switching database. Figure 1-7 outlines the interaction between these applications and the label-switching matrix.

Figure 1-7 *Various MPLS Applications and Their Interactions*



Every MPLS application has the same set of components as the IP routing application:

- A database defining the Forward Equivalence Classes (FECs) table for the application (the IP routing table in an IP routing application)
- Control protocols that exchange the contents of the FEC table between the LSRs (IP routing protocols or static routing in an IP routing application)
- Control process that performs label binding to FECs and a protocol to exchange label bindings between LSRs (TDP or LDP in an IP routing application)
- Optionally, an internal database of FEC-to-label mapping (Label Information Base in an IP routing application)

Each application uses its own set of protocols to exchange FEC table or FEC-to-label mapping between nodes. Table 1-2 summarizes the protocols and the data structures.

The next few chapters cover the use of MPLS in IP routing; Part II, "MPLS-based Virtual Private Networks," covers the Virtual Private Networking application.

Table 1-2 *Control Protocols Used in Various MPLS Applications*

Application	FEC Table	Control Protocol Used to Build FEC Table	Control Protocol Used to Exchange FEC-to-Label Mapping
IP routing	IP routing table	Any IP routing protocol	Tag Distribution Protocol (TDP) or Label Distribution Protocol (LDP)
Multicast IP routing	Multicast routing table	PIM	PIM version 2 extensions
Application	**FEC Table**	**Control Protocol Used to Build FEC Table**	**Control Protocol Used to Exchange FEC-to-Label Mapping**
VPN routing	Per-VPN routing table	Most IP routing protocols between service provider and customer, Multiprotocol BGP inside the service provider network	Multiprotocol BGP
Traffic engineering	MPLS tunnels definition	Manual interface definitions, extensions to IS-IS or OSPF	RSVP or CR-LDP
MPLS Quality of Service	IP routing table	IP routing protocols	Extensions to TDP LDP

Summary

Traditional IP routing has several well-known limitations, ranging from scalability issues to poor support of traffic engineering and poor integration with Layer 2 backbones already existing in large service provider networks. With the rapid growth of the Internet and the establishment of IP as the Layer 3 protocol of choice in most environments, the drawbacks of traditional IP routing became more and more obvious.

MPLS was created to combine the benefits of connectionless Layer 3 routing and forwarding with connection-oriented Layer 2 forwarding. MPLS clearly separates the control plane, where Layer 3 routing protocols establish the paths used for packet forwarding, and the data plane, where Layer 2 label switched paths forward data packets across the MPLS infrastructure. MPLS also simplifies per-hop data forwarding, where it replaces the Layer 3 lookup function performed in traditional routers with simpler label swapping. The simplicity of data plane packet forwarding and its similarity to existing Layer 2 technologies enable traditional WAN equipment (ATM or Frame Relay switches) to be redeployed as MPLS nodes (supporting IP routing in the control plane) just with software upgrades to their control plane.

The control component in the MPLS node uses its internal data structure to identify potential traffic classes (also called Forward Equivalence Classes). A protocol is used between control components in MPLS nodes to exchange the contents of the FEC database and the FEC-to-label mapping. The FEC table and FEC-to-label mapping is used in Edge-LSRs to label ingress packets and send them into the MPLS network. The Label Forwarding Information Base (LFIB) is built within each MPLS node based on the contents of the FEC tables and the FEC-to-label mapping exchanged between the nodes. The LFIB then is used to propagate labeled packets across the MPLS network, similar to the function performed by an ATM switching matrix in the ATM switches.

The MPLS architecture is generic enough to support other applications besides IP routing. The simplest additions to the architecture are the IP multicast routing and quality of service extensions. The MPLS connection-oriented forwarding mechanism together with Layer 2 label-based look ups in the network core also has enabled a range of novel applications, from Traffic Engineering to real peer-to-peer Virtual Private Networks.