



MPLS, Part II

Peter J. Welcher

Introduction

Last month we started looking at MPLS, Multiprotocol Label Switching. This technology is very hot among large Enterprises and Service Providers. MPLS looks like the best approach for simplified implementation of very large IP infrastructures, be they high speed IP directly over SONET, IP over ATM, and (in the near future) IP over optical. We went into some of this in the previous article, which can be found at:

- <http://www.netcraftsmen.net/welcher/papers/mplsintro.html>

You may also wish to review QoS features by looking at:

- <http://www.netcraftsmen.net/welcher/papers/qos1.html>
- <http://www.netcraftsmen.net/welcher/papers/qos2.html>
- <http://www.netcraftsmen.net/welcher/papers/qos3.html>
- <http://www.netcraftsmen.net/welcher/papers/newqos121.html>

In this article I'd like to follow up with "fancier" MPLS, namely Class of Service with MPLS, and Traffic Engineering (TE). That's a lot for one article! My reasoning is that many readers may want to know a little more about MPLS, but relatively few actually need all the details. I'm sure you'll send me email if I'm wrong about that.

By the way, if you're wondering about what equipment to try this on, it looks as if 12.1(3) T supports MPLS in 2600 routers (although the feature matrix doesn't show this). So MPLS over IP should be fairly do-able, equipment-wise.

A caution: this is all fairly new stuff, I do not have equipment available to test it with (nor time), and am piecing together information from various sources. Thus the configurations are my best effort but are not guaranteed accurate. They are intended to give you a feel for what's involved in configuring the features discussed.

Configuring Basic MPLS

Turning on basic MPLS is pretty simple:

```
! if you're on a platform supporting dCEF:
ip cef distributed

! turn on MPLS tag distribution
```

```
tag-switching advertise-tags

! enable MPLS on appropriate interface(s) :
interface e0/1
tag-switching ip
```

You should look at

http://www.cisco.com/univercd/cc/td/doc/product/software/ios121/121cgcr/switch_c/xcprt4/xchtagc.l

for details on how to enable MPLS incrementally. You can use an access list with the advertise-tags command, to specify which networks to advertise labels for, presumably those that can be reached via MPLS. You can even control which peer LSRs to advertise which prefixes to.

MPLS Class of Service

Since the marketing and interest in MPLS is tied up with ATM coexisting with IP, the question of providing Quality of Service (QoS) always comes up. The focus in MPLS is more on differentiated Classes of Service than on ATM-like QoS, although with Traffic Engineering features, MPLS seems to come a long way towards IP-style QoS.

Right now, the Cisco CoS features used for MPLS are CAR or CBWFQ, WRED, and WFQ.

We start by using CAR or CBWFQ (or a couple of other techniques, see the QoS articles) to classify or recognize traffic at the edge of the network. These techniques also let us mark the traffic, setting the 3 IP Precedence or 6 DSCP bits in the IP Type of Service field. Recall that marking allows downstream (core) devices to provide appropriate service to the packet without having to delve as closely into headers to figure out what service the packet deserves.

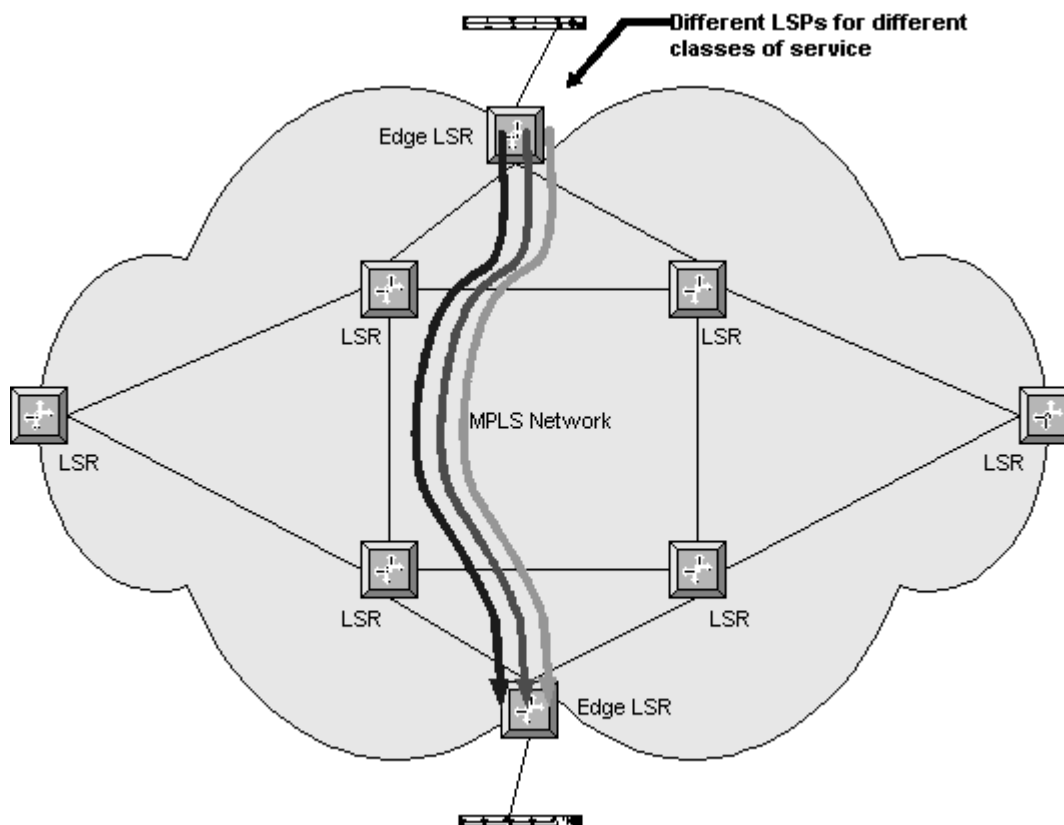
We also configure WRED or WFQ to provide differentiated service based on IP Precedence (or DSCP) in the downstream (core) routers. These queue management and scheduling techniques can be applied whether or not MPLS is in effect, if we're operating MPLS over IP. The IP Precedence information determines the weights to be used (the 'W' in WRED or WFQ). Higher IP Precedence gets preferential treatment.

MPLS comes into the picture in two possible ways. One is by copying IP Precedence bits to the MPLS header (if desired). This MPLS header is used for MPLS over IP and has a field for such CoS information, the EXP field (3 bits). The second way MPLS can deal with CoS is by storing Precedence information as part of the Label Information Base (LIB). Each level of precedence is assigned a different Label Switch Path, so the label can be thought of as implicitly specifying the precedence. (If a Label Switch Router needs to know the precedence, it can look it up in the LIB.)

So when a frame arrives at a LSR, the label is used to determine outbound interface and new label, but the precedence or EXP field is then used to determine queuing treatment.

On ATM LSRs, the same thing happens. We're dealing with a Label Virtual Circuit (LVC) for our Label Switch Path. The LIB determines outgoing interface, which happens to be an ATM interface. WFQ and WRED can then be applied on the outgoing ATM interface, along with WEPD (Weighted Early Packet Discard).

With a non-MPLS ATM core, the edge LSRs are interconnected by ATM PVCs through the core ATM switches. WFQ and WRED can be applied on a per-VC basis. The BPX 8650 also allows you to use different PVCs for different classes of service.



Configuring MPLS CoS

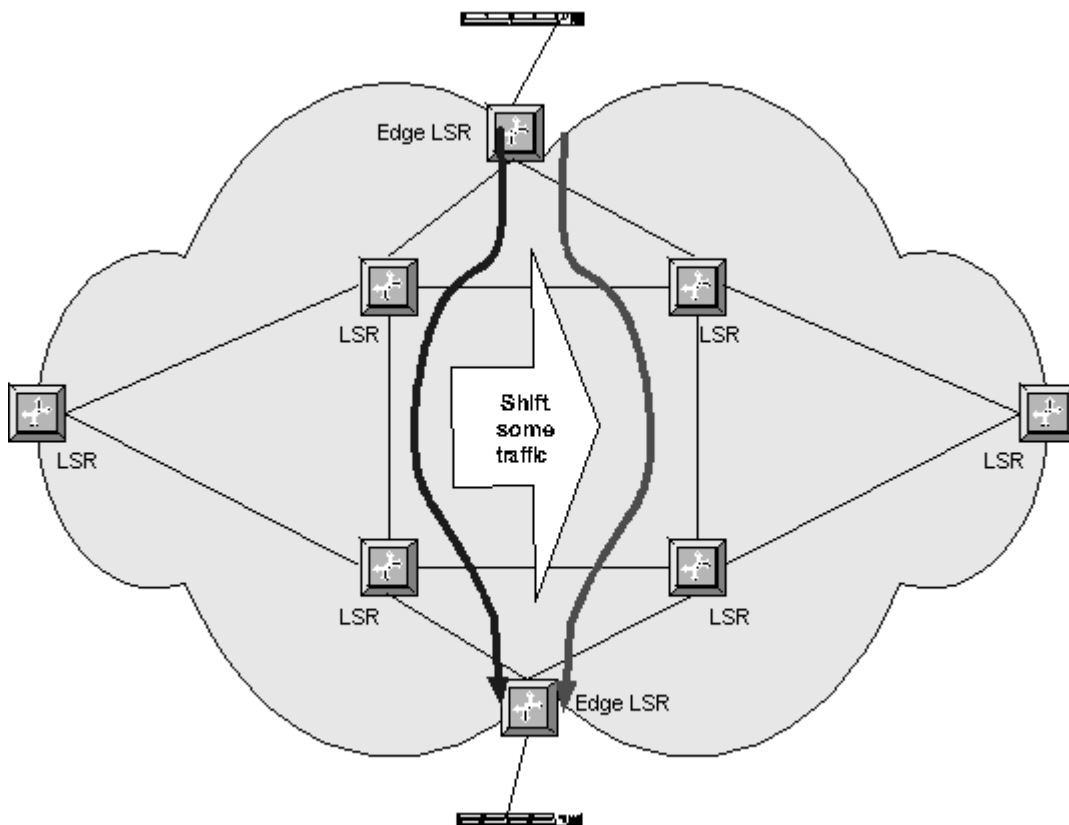
To use multiple VCs for MPLS Cos on an ATM interface, configure:

```
interface e0/1
  tag-switching atm multi-vc
  tag-switching ip
```

This creates four VCs for each MPLS destination. An alternative is to use fewer label VCs by configuring CoS mapping. See the documentation (basically, the above URL) for details and alternatives.

MPLS Traffic Engineering

The idea of MPLS Traffic Engineering is to use unidirectional tunnels to shift traffic off one path and onto another. The tunnels can be statically or automatically determined by the LSRs. Multiple tunnels can be used for load sharing when a traffic flow is too large for a single path.



Although the figure shows edge to edge tunnels, TE tunnels can be shorter. They can be used by a Service Provider to shift traffic off an overloaded trunk, until more capacity can be added.

The tunnel mechanism works because we can stack up the labels applied to IP packets. That is, additional labels are applied temporarily, to the outside of the packet and existing label, to shunt traffic into the tunnel. The tunnel LSP is followed until the end of the tunnel, where the outermost label is popped off. At that point the packet resumes following the original LSP to its destination.

A link state protocol (IS-IS or OSPF) is used with enhanced link state advertisements to track network capacity and to ensure that the tunnel does not create a routing loop. The actual signaling for dynamic tunnel establishment is based on RSVP, which acts to reserve bandwidth on a link.

The following example shows all of these factors at work. It sets up an explicit tunnel (where we statically specify the path) with a dynamic backup tunnel. This is a configuration snippet from the LSR at the entrance to the tunnel (top of the picture).

```
mpls traffic-eng tunnels

interface fast 0/0
 ip address 10.1.1.1 255.255.255.0
 mpls traffic-eng tunnels
 ip rsvp bandwidth 10000

interface tunnel 1
 tunnel destination 10.3.3.3
 tunnel mode mpls traffic-eng
 tunnel mpls traffic-eng path-option 1 explicit name mytunnel
 tunnel mpls traffic-eng bandwidth 1000
```

```
tunnel mpls traffic-eng path-option 2 dynamic

ip explicit-path name mytunnel
  next-address 10.1.2.1
  next-address 10.1.10.1
  next-address 10.3.3.3
```

You also would have to enable tunnels on routers and interfaces the tunnel might traverse:

```
mpls traffic-eng tunnels

interface fast 0/0
  mpls traffic-eng tunnels

interface fast 1/0
  mpls traffic-eng tunnels
```

For the dynamic path establishment to work, we would also need to configure IS-IS for MPLS Traffic Engineering, and specify which traffic is to use the tunnel. The traffic to go through this tunnel is that exiting the BGP Autonomous System at router 10.5.5.5.

```
ip router isis

router isis
  net 47.0000.0012.3456.00
  is-type level-1
  mpls traffic-eng router-id loopback0
  mpls traffic-eng level-1
  metric-style wide

router traffic-engineering
  traffic-engineering filter 60 egress 10.5.5.5 255.255.255.255
  traffic-engineering route 60 tunnel 1
```

The **metric-style wide** command allows ISIS to track the additional routing metric information needed for Traffic Engineering. There is a routing protocol migration issue here and you should read all the relevant documentation before attempting this in a production network! See:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios121/121cgcr/switch_c/xcprt4/xcdtagov

Summary

I had somewhat hoped to discuss MPLS-based VPNs as well, since they're actually in use and are pretty nifty. We'll have to leave them for a possible future article.

The **ping tag** command (which I couldn't find in the documentation) uses an MPLS path to ping. If there is no MPLS path it fails. This is convenient for testing edge to edge path connectivity in an MPLS network.

Dr. Peter J. Welcher (CCIE #1773, CCSI #94014) is a Senior Consultant with Chesapeake NetCraftsmen. NetCraftsmen is a high-end consulting firm and Cisco Premier Partner dedicated to

quality consulting and knowledge transfer. NetCraftsmen has nine CCIE's, with expertise including large network high-availability routing/switching and design, VoIP, QoS, MPLS, network management, security, IP multicast, and other areas. See <http://www.netcraftsmen.net/> for more information about NetCraftsmen. Pete's links start at <http://www.netcraftsmen.net/welcher> . New articles will be posted under the Articles link. Questions, suggestions for articles, etc. can be sent to pjw@netcraftsmen.net .

9/4/2000

Copyright (C) 2000, Peter J. Welcher