ECSE-6660
Label Switching and MPLS

http://www.pde.rpi.edu/
Or
http://www.ecse.rpi.edu/Homepages/shivkuma/

Shivkumar Kalyanaraman
Rensselaer Polytechnic Institute
shivkuma@ecse.rpi.edu

Based in part on slides from Prof. Raj Jain (OSU),
Kireeti Kompella, Juniper networks,
Peter Ashwood-Smith and Bilel Jamoussi (Nortel Networks),
Overview

- IP-over-ATM to MPLS: History of IP Switching
- MPLS: generalization of labels, de-coupling of control plane
- Label distribution/setup protocols: RSVP, LDP
- Introduction to Traffic Engineering
IP: “Best-Effort Philosophy”

- Well architected, not necessarily worked out in detail
- Realization: can’t predict the future
- Architectural decisions:
  - Make it reasonable
  - Make it flexible
  - Make it extensible

stuff above
transport
network
stuff below
IP Control Plane Evolution

- Again, just good enough (best-effort) …
  - But again, flexible, extensible
- Distance Vector routing was fine for quite a while
  - Just in time, along came link state (OSPF and IS-IS)
- Now a burning question in OSPF/IS-IS is:
  - Convergence “in a few seconds” is not good enough?
  - See NANOG June 2002 for interesting videos and papers on how to fix LS-routing for fast convergence

- **Goal:** “Business” IP for service providers…
  - Make me money – new services, GoS
  - Don’t lose me money – uptime, SLAs
  - OSPF/BGP not originally designed to support QoS or multiple services (eg: VoIP, VPNs)
ATM – Perfectionist’s Dream

- Connection-oriented
- Does everything and does it well
- Anticipated all future uses and factored them in
- Philosophical mismatch with IP
Overlay Model for IP-over-ATM Internetworking

- Goal: Run IP over ATM core networks
- Why? ATM switches offered performance, predictable behavior and services (CBR, VBR, GFR...)
- ISPs created “overlay” networks that presented a virtual topology to the edge routers in their network
- Using ATM virtual circuits, the virtual network could be reengineered without changing the physical network
- Benefits
  - Full traffic control
  - Per-circuit statistics
  - More balanced flow of traffic across links
Overlay Model (Contd)

- ATM core ringed by routers
- PVCs overlaid onto physical network
**Issue 1: Mapping IP data-plane to ATM: Address Resolution Woes!**

- A variety of *server-based* address resolution servers:
  - ATMARP (RFC 1577), LANE server, BUS server, MPOA server, NHRP server....
  - Use of separate pt-pt and pt-mpt VCs with servers
  - Multiple servers + backup VCs to them needed for fault tolerance
  - Separate servers needed in every LOGICAL domain (eg: LIS)
- Mismatch between the notion of IP subnet and ATM network sizing
  - *Cut-through forwarding* between nodes on same ATM network hard to achieve!
Issue 2: Mapping IP control-plane (eg: OSPF) to ATM

- Basic OSPF assumes that subnets are pt-pt or offer broadcast capability.
- ATM is a Non-Broadcast Multiple Access (NBMA) media
  - NBMA “segments” support multiple “routers” with pt-pt VCs but do not support data-link broadcast/mcast capability
  - Each VC is costly => setting up full mesh for OSPF Hello messages is prohibitively expensive!
- Two “flooding adjacency” models in OSPF:
  - Non-Broadcast Multiple Access (NBMA) model
  - Point-to-Multipoint (pt-mpt) Model
- Different tradeoffs…
Partial Mesh: **NBMA model**

1. **Neighbor discovery**: manually configured

2. **Dijkstra SPF** views NBMA as a full mesh!
Partial Mesh: *pt-mpt model*

*Figure 5.10* Turning the partial mesh of Figure 5.9 into a Point-to-MultiPoint subnet.
Key assumption in NBMA model:
- Each router on the subnet can communicate with every other (same as IP subnet model)
- But this requires a “full mesh” of expensive PVCs at the lower layer!
- Many organizations have a hub-and-spoke PVC setup, a.k.a. “partial mesh”
- Conversion into NBMA model requires multiple IP subnets, and complex configuration (see fig on next slide)
- OSPF’s pt-mpt subnet model breaks the rule that two routers on the same network must be able to talk directly
- Can turn partial PVC mesh into a single IP subnet
Instead of sending a separate router-LSA for each router, one “designated router” can create a network-LSA for the subnet.
OSPF Designated Router (DR): NBMA Case

- One router elected as a designated router (DR)
- Each router in subnet maintains “flooding adjacency” with the DR, i.e., sends acks of LSAs to DR
- DR informs each router of other routers on LAN
- DR generates the network-LSA on subnet’s behalf after synchronizing with all routers
- Complex election protocol for DR in case of failure
DR and BDR in OSPF NBMA model

- In NBMA model:
  - **DR and BDR** only maintain VCs and Hellos with *all* routers on NBMA
  - Flooding in NBMA always goes through DR
  - Multicast not available to optimize LSA flooding.
  - DR generates network-LSA
Summary: IP-to-ATM Overlay Model Drawbacks

- **IP-to-ATM: control-plane** mapping issues
  - Need a *full mesh of ATM PVCs* for mapping IP routing
  - Both NBMA and Pt-Mpt mapping models have drawbacks

- **IP-to-ATM: data-plane** mapping issues
  - *Address resolution* (eg: LANE, RFC 1577, MPOA, NHRP) requires a complex distributed server and multicast VC infrastructure
  - *Segmentation-and-Reassembly (SAR)* of IP packets into ATM cells can have a multiplier-effect on performance even if one cell in a packet is lost
    - ATM SAR has trouble scaling to OC-48 and OC-192 speeds
    - Packet-over-SONET (POS) emerged as an alternative at the link layer
  - ATM + AAL5 *overhead* (20%) deemed excessive
Re-examining Basics: Routing vs Switching

- **Routing**: Based on address lookup. Max prefix match.
  - Search Operation
  - Complexity \( \approx O(\log_2 n) \)

- **Switching**: Based on circuit numbers
  - Indexing operation
  - Complexity \( O(1) \)
  - Fast and Scalable for large networks and large address spaces

- These distinctions apply on all datalinks: ATM, Ethernet, SONET
IP Routing vs IP Switching

On ATM networks:

- IP routers use IP addresses
  ⇒ Reassemble IP datagrams from cells
- IP Switches use ATM Virtual circuit numbers
  ⇒ Switch cells
  ⇒ Do not need to reassemble IP datagrams
Caveat: one cares about combining the best of both worlds only for large ISP networks that need both features!

Note: the “hybrid” also happens to be a solution that bypasses IP-over-ATM mapping woes!
History: Ipsilon’s IP Switching: Concept

- Developed by Ipsilon
- Routing software in every ATM switch in the network
- Initially, packets are reassembled by the routing software and forwarded to the next hop
- Long term flows are transferred to separate VCs. Mapping of VCI in the switch \(\Rightarrow\) No reassembly

**Hybrid:** IP routing (control plane) + ATM switching (data plane)
Ipsilon’s IP Switching

ATM VCs setup when new IP “flows” seen, i.e., “data-driven” VC setup

- Flow-oriented traffic: FTP, Telnet, HTTP, Multimedia
- Short-lived Traffic: DNS query, SMTP, NTP, SNMP, request-response
  Ipsilon claimed that 80% of packets and 90% of bytes are flow-oriented.
- Ipsilon claimed their Generic Switch Management Protocol (GSMP) to be 2000 lines, and Ipsilon Flow Management Protocol (IFMP) to be only 10,000 lines of code
- Runs as added software on an ATM switch
Issues with Ipsilon’s IP switching

- VCI field is used as ID.
  - VPI/VCI change at switch
  - Must run on every ATM switch
  - non-IP switches not allowed between IP switches
  - Subnets limited to one switch
- Cannot support VLANs
- Scalability: Number of VC ≥ Number of flows.
  - VC Explosion. 1000 setups/sec.
- Quality of service determined implicitly by the flow class or by RSVP
- ATM Only
Tag Switching

- Proposed by CISCO
- Similar to VLAN tags
- Tags can be explicit or implicit L2 header

L2 Header | Tag

- Ingress router/host puts a tag. Exit router strips it off.

Key difference: tags can be setup in the background using IP routing protocols (i.e. control-driven VC setup)
Alphabet Soup!

- CSR Cell Switched Router
- ISR Integrated Switch and Router
- LSR Label Switching Router
- TSR Tag Switching Router
- Multi layer switches, Swoters
- DirectIP
- FastIP
- PowerIP

MPLS working group in IETF was formed to reach a common standard
MPLS Broad Concept: Route at Edge, Switch in Core
MPLS Terminology

- **LDP**: Label Distribution Protocol
- **LSP**: Label Switched Path
- **FEC**: Forwarding Equivalence Class
- **LSR**: Label Switching Router
- **LER**: Label Edge Router (Useful term not in standards)

MPLS is “multi-protocol” both in terms of the protocols it supports ABOVE it and BELOW it in the protocol stack!
MPLS Header

- IP packet is **encapsulated** in MPLS header and sent down LSP

- IP packet is **restored** at end of LSP by egress router
  - TTL is adjusted by default
MPLS Label Stack Concept

- Labels = Explicit or implicit L2 header
- TTL = Time to live
- Exp = Experimental
- SI = Stack indicator

![Diagram of MPLS Label Stack]

Allows nested tunnels, that are opaque, i.e. do not know or care what protocol data they carry (a.k.a multi-protocol)
### MPLS Header

<table>
<thead>
<tr>
<th>Label</th>
<th>EXP</th>
<th>S</th>
<th>TTL</th>
</tr>
</thead>
</table>

- **Label**: Used to match packet to LSP
- **Experimental bits**: Carries packet queuing priority (CoS)
- **Stacking bit**: can build “stacks” of labels
  - Goal: nested tunnels!
- **Time to live**: Copied from IP TTL
Multi-protocol operation

The abstract notion of a “label” can be mapped to multiple circuit- or VC-oriented technologies!

- **ATM** - label is called VPI/VCI and travels with cell.
- **Frame Relay** - label is called a DLCI and travels with frame.
- **TDM** - label is called a timeslot its implied, like a lane.
- **X25** - a label is an LCN
- Proprietary labels: TAG (in tag switching) etc..
- **Frequency or Wavelength substitution** where “label” is a light frequency/wavelength? (idea in G-MPLS)
MPLS Encapsulation is specified over various media types. Top labels may use existing format, lower label(s) use a new “shim” label format.
MPLS Encapsulation - ATM

ATM LSR constrained by the cell format imposed by existing ATM standards

ATM Header Format

<table>
<thead>
<tr>
<th>VPI</th>
<th>VCI</th>
<th>PT</th>
<th>CLP</th>
<th>HEC</th>
</tr>
</thead>
</table>

ATM Header Payload

ATM SAR

Generic Label Encap. (PPP/LAN format)

Network Layer Header and Packet (eg. IP)

AAL5 Trailer

48 Bytes

ATM VPI (Tunnel) Label

Option 1

Option 2

Option 3

ATM Header

5 Octets

AAL 5 PDU Frame (nx48 bytes)

• Top 1 or 2 labels are contained in the VPI/VCI fields of ATM header
  - one in each or single label in combined field, negotiated by LDP
• Further fields in stack are encoded with ‘shim’ header in PPP/LAN format
  - must be at least one, with bottom label distinguished with ‘explicit NULL’
• TTL is carried in top label in stack, as a proxy for ATM header (that lacks TTL)
MPLS Encapsulation - Frame Relay

- Current label value carried in DLCI field of Frame Relay header
- Can use either 2 or 4 octet Q.922 Address (10, 17, 23 bytes)
- Generic encapsulation contains n labels for stack of depth n
  - top label contains TTL (which FR header lacks), ‘explicit NULL’ label value

DLCI Size = 10, 17, 23 Bits
MPLS Encapsulation: PPP & LAN Data Links

- Network layer must be inferable from value of bottom label of the stack
- TTL must be set to the value of the IP TTL field when packet is first labelled
- When last label is popped off stack, MPLS TTL to be copied to IP TTL field
- Pushing multiple labels may cause length of frame to exceed layer-2 MTU
  - LSR must support “Max. IP Datagram Size for Labelling” parameter
  - any unlabelled datagram greater in size than this parameter is to be fragmented

MPLS on PPP links and LANs uses ‘Shim’ Header Inserted Between Layer 2 and Layer 3 Headers
MPLS Forwarding: Example

- An IP packet destined to 134.112.1.5/32 arrives in SF
- San Francisco has route for 134.112/16
- Next hop is the LSP to New York
MPLS Forwarding Example

- San Francisco pre-pends MPLS header onto IP packet and sends packet to first transit router in the path.
MPLS Forwarding Example

- Because the packet arrived at Santa Fe with an MPLS header, Santa Fe forwards it using the MPLS forwarding table.
- MPLS forwarding table derived from mpls.0 switching table.
MPLS Forwarding Example

- Packet arrives from penultimate router with label 0
- Egress router sees label 0 and strips MPLS header
- Egress router performs standard IP forwarding decision
Label Setup/Signaling: MPLS Using IP Routing Protocols

- Destination based forwarding tables as built by OSPF, IS-IS, RIP, etc.
Regular IP Forwarding

IP destination address unchanged in packet header!
MPLS Label Distribution

<table>
<thead>
<tr>
<th>Intf In</th>
<th>Label In</th>
<th>Dest</th>
<th>Intf Out</th>
<th>Label Out</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>0.50</td>
<td>47.1</td>
<td>1</td>
<td>0.40</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Intf In</th>
<th>Label In</th>
<th>Dest</th>
<th>Intf Out</th>
<th>Label Out</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>0.40</td>
<td>47.1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

Mapping: 0.40
Request: 47.1
Mapping: 0.50
Request: 47.1

Intf In  Dest  Intf Out  Label Out
3  47.1  1  0.50

47.1 1
47.2 2
Label Switched Path (LSP)
- A Vanilla LSP is actually part of a tree from every source to that destination (unidirectional).
- Vanilla LDP builds that tree using existing IP forwarding tables to route the control messages.
Explicitly Routed (ER-) LSP

ER-LSP follows route that source chooses. In other words, the control message to establish the LSP (label request) is source routed.
Explicitly Routed (ER-) LSP Contd

<table>
<thead>
<tr>
<th>Inf In</th>
<th>Dest In</th>
<th>Inf Out</th>
<th>Label Out</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>47.1.1</td>
<td>2</td>
<td>1.33</td>
</tr>
<tr>
<td>3</td>
<td>47.1</td>
<td>1</td>
<td>0.50</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Inf In</th>
<th>Label In</th>
<th>Dest</th>
<th>Inf Out</th>
<th>Label Out</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>0.50</td>
<td>47.1</td>
<td>1</td>
<td>0.40</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Inf In</th>
<th>Label In</th>
<th>Dest</th>
<th>Inf Out</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>0.40</td>
<td>47.1</td>
<td>1</td>
</tr>
</tbody>
</table>

IP 47.1.1.1

47.3 3

2

2

IP 47.1.1.1

3

47.2
ER LSP - advantages

- Operator has routing flexibility (policy-based, QoS-based)
- Can use routes other than shortest path
- Can compute routes based on constraints in exactly the same manner as ATM based on distributed topology database. (traffic engineering)
ER LSP - discord!

- Two signaling options proposed in the standards: CR-LDP, RSVP extensions:
  - CR-LDP = LDP + Explicit Route
  - RSVP ext = Traditional RSVP + Explicit Route + Scalability Extensions
- Not going to be resolved any time soon, market will probably have to resolve it.
Traffic Engineering

- **TE**: “…that aspect of Internet network engineering dealing with the issue of performance evaluation and *performance optimization of operational IP networks* …”

- **Two abstract sub-problems**:
  1. Define a *traffic aggregate* (e.g., OC-, T-carrier hierarchy, or ATM PVCs)
  2. Map the traffic aggregate to an explicitly setup path

- Cannot do this in OSPF or BGP-4 today!

- OSPF and BGP-4 offer only a *SINGLE* path!

```
A  B  C  D
|   |   |   |
1  1  1  4

Links AB and BD are overloaded
Links AC and CD are overloaded
```
Why not TE with OSPF/BGP?

- Internet connectionless routing protocols designed to find only one route (path)
  - The “connectionless” approach to TE is to “tweak” (i.e. change) link weights in IGP (OSPF, IS-IS) or EGP (BGP-4) protocols
  - Assumptions: Quasi-static traffic, knowledge of demand matrix

- **Limitations:**
  - Performance is fundamentally limited by the single shortest/policy path nature:
    - All flows to a destination prefix mapped to the same path
  - Desire to map traffic to different route (eg: for load-balancing reasons) => the single default route **MUST** be changed
  - Changing parameters (eg: OSPF link weights) changes routes **AND** changes the traffic mapped to the routes
  - Leads to *extra control traffic* (eg: OSPF floods or BGP-4 update message), *convergence problems* and *routing instability*!

- **Summary:** Traffic mapping **coupled** with route availability in OSPF/BGP!
  - MPLS **de-couples** traffic trunking from path setup
Traffic Engineering w/ MPLS (Step I)

- Engineer *unidirectional* paths through your network *without* using the IGP’s shortest path calculation.
Traffic Engineering w/ MPLS (Part II)

- IP prefixes (or traffic aggregates) can now be bound to MPLS Label Switched Paths (LSPs)
Traffic Aggregates: Forwarding Equivalence Classes

- **FEC** = “A subset of packets that are all treated the same way by a router”
- The concept of FECs provides for a great deal of flexibility and scalability
- In conventional routing, a packet is assigned to a FEC at each hop (i.e. L3 look-up), in MPLS it is only done once at the network ingress
Signaled TE Approach (eg: MPLS)

- **Features:**
  - In MPLS, the choice of a route (and its setup) is orthogonal to the problem of traffic mapping onto a route.
  - Signaling maps global IDs (addresses, path-specification) to local IDs (labels).
  - FEC mechanism for defining traffic aggregates, label stacking for multi-level opaque tunneling.

- **Issues:**
  - Requires extensive upgrades in the network.
  - Hard to inter-network beyond area boundaries.
  - Very hard to go beyond AS boundaries (even in same organization).
  - Impossible for inter-domain routing across multiple organizations => inter-domain TE has to be connectionless.
# Hop-by-Hop vs. Explicit Routing

<table>
<thead>
<tr>
<th>Hop-by-Hop Routing</th>
</tr>
</thead>
<tbody>
<tr>
<td>Distributes routing of control traffic</td>
</tr>
<tr>
<td>Builds a set of trees either fragment by fragment like a random fill, or backwards, or forwards in organized manner.</td>
</tr>
<tr>
<td>Reroute on failure impacted by convergence time of routing protocol</td>
</tr>
<tr>
<td>Existing routing protocols are destination prefix based</td>
</tr>
<tr>
<td>Difficult to perform traffic engineering, QoS-based routing</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Explicit Routing</th>
</tr>
</thead>
<tbody>
<tr>
<td>Source routing of control traffic</td>
</tr>
<tr>
<td>Builds a path from source to dest</td>
</tr>
<tr>
<td>Requires manual provisioning, or automated creation mechanisms.</td>
</tr>
<tr>
<td>LSPs can be ranked so some reroute very quickly and/or backup paths may be pre-provisioned for rapid restoration</td>
</tr>
<tr>
<td>Operator has routing flexibility (policy-based, QoS-based,</td>
</tr>
<tr>
<td>Adapts well to traffic engineering</td>
</tr>
</tbody>
</table>

Explicit routing shows great promise for traffic engineering.
RSVP: “Resource reSerVation Protocol”

- A generic QoS signaling protocol
- An Internet control protocol
  - Uses IP as its network layer
- Originally designed for host-to-host
- Uses the IGP to determine paths
- RSVP is not
  - A data transport protocol
  - A routing protocol
- RFC 2205
Recall: Signaling ideas

- Classic scheme: sender initiated
- SETUP, SETUP_ACK, SETUP_RESPONSE
- Admission control
- Tentative resource reservation and confirmation
- Simplex and duplex setup; no multicast support
RSVP: Internet Signaling

- Creates and maintains distributed reservation state
- **De-coupled** from routing & also to support **IP multicast model**:
  - Multicast trees setup by routing protocols, not RSVP (unlike ATM or telephony signaling)
- Key features of RSVP:
  - **Receiver-initiated**: scales for multicast
  - **Soft-state**: reservation times out unless refreshed
- Latest paths discovered through “PATH” messages (forward direction) and used by **RESV** mesgs (reverse direction).
  - Again dictated by needs of de-coupling from IP routing and to support IP multicast model
RSVP Path Signaling Example

- Signaling protocol sets up path from San Francisco to New York, reserving bandwidth along the way.
RSVP Path Signaling Example

- Once path is established, signaling protocol assigns label numbers in reverse order from New York to San Francisco.
Call Admission

- Session must first declare its QoS requirement and characterize the traffic it will send through the network
- **R-spec**: defines the QoS being requested
- **T-spec**: defines the traffic characteristics
- A signaling protocol is needed to carry the R-spec and T-spec to the routers where reservation is required; RSVP is a leading candidate for such signaling protocol
Call Admission

- **Call Admission**: routers will admit calls based on their R-spec and T-spec and base on the current resource allocated at the routers to other calls.

1. **Request**: specify
   - traffic (Tspec)
   - guarantee (Rspec)

2. **Element considers**
   - unreserved resources
   - required resources

3. **Reply**: whether or not request can be satisfied.
Summary: Basic RSVP Path Signaling

- Reservation for simplex (unidirectional) flows
- Ingress router initiates connection
- “Soft” state
  - Path and resources are maintained dynamically
  - Can change during the life of the RSVP session
- Path message sent downstream
- Resv message sent upstream
MPLS Extensions to RSVP

- Path and Resv message objects
  - Explicit Route Object (ERO)
  - Label Request Object
  - Label Object
  - Record Route Object
  - Session Attribute Object
  - Tspec Object
- For more detail on contents of objects:
  daft-ietf-mpls-rsvp-lsp-tunnel-04.txt

Extensions to RSVP for LSP Tunnels
Explicit Route Object

- Used to specify the explicit route RSVP Path messages take for setting up LSP.
- Can specify loose or strict routes:
  - Loose routes rely on routing table to find destination.
  - Strict routes specify the directly-connected next router.
- A route can have both loose and strict components.
ERO: Strict Route

- Next hop must be directly connected to previous hop

![Diagram of network with strict route requirements]

**ERSO**

- B strict; C strict; E strict; D strict; F strict;

**Ingress LSR**

**Egress LSR**
ERO: Loose Route

- Consult the routing table at each hop to determine the best path: similar to IP routing option concept.
ER0: Strict/Loose Path

- Strict and loose routes can be mixed
Label Objects

- **Label Request Object**
  - Added to PATH message at ingress LSR
  - Requests that each LSR provide label to upstream LSR

- **Label Object**
  - Carried in RESV messages along return path upstream
  - Provides label to upstream LSR
Record Route Object—PATH Message

- Added to PATH message by ingress LSR
- Adds outgoing IP address of each hop in the path
  - In downstream direction

- Loop detection mechanism
  - Sends “Routing problem, loop detected” PathErr message
  - Drops PATH message
Session Attribute Object

- Added to PATH message by ingress router
- **Controls LSP**
  - Priority
  - Preemption
  - Fast-reroute
- **Identifies session**
  - ASCII character string for LSP name
Adjacency Maintenance—Hello Message

- New RSVP extension: leverage RSVP for hellos!
  - Hello message
  - Hello Request
  - Hello Acknowledge

- Rapid node to node failure detection
  - Asynchronous updates
  - 3 second default update timer
  - 12 second default dead timer
Path Maintenance — Refresh Messages

- Maintains reservation of each LSP
- Sent every 30 seconds by default
- Consists of PATH and RESV messages
RSVP Message Aggregation

- Bundles up to 30 RSVP messages within single PDU
- Controls
  - Flooding of PathTear or PathErr messages
  - Periodic refresh messages (PATH and RESV)
- Enhances protocol efficiency and reliability
- Disabled by default
Signaled vs Constrained LSPs

- Common Features
  - Signaled by RSVP
  - MPLS labels automatically assigned
  - Configured on ingress router only

- Signaled LSPs
  - CSPF not used (i.e. normal IP routing is used)
  - User configured ERO handed to RSVP for signaling
  - RSVP consults routing table to make next hop decision

- Constrained LSPs
  - CSPF used
  - Full path computed by CSPF at ingress router
  - Complete ERO handed to RSVP for signaling
Constrained Shortest Path First Algorithm

- Modified “shortest path first” algorithm
- Finds shortest path based on IGP metric while satisfying additional QoS constraints
- Integrates TED (Traffic Engineering Database)
  - IGP topology information
  - Available bandwidth
  - Link color
- Modified by administrative constraints
  - Maximum hop count
  - Bandwidth
  - Strict or loose routing
  - Administrative groups
Computing the ERO

- Ingress LSR passes user defined restrictions to CSPF
  - Strict and loose hops
  - Bandwidth constraints
  - Admin Groups
- CSPF algorithm
  - Factors in user defined restrictions
  - Runs computation against the TED
  - Determines the shortest path
- CSPF hands full ERO to RSVP for signaling
Summary: Key Benefits of MPLS

- **Goal:** Low-overhead virtual circuits for IP
  - Originally designed to make routers faster by leveraging ATM switch cores (bypasses IP-over-ATM overlay problems)
  - Fixed label lookup faster than longest match used by IP routing
    - Caveat: Not true anymore!
  - IP forwarding has broken terabit/s speeds through innovative data-structures (next class)!
  - PPP-over-SONET (POS) provides a link layer!

- **Value of MPLS** is now purportedly in “traffic engineering”
  - Same forwarding mechanism can support **multiple new services** (eg: VoIP, VPNs etc)
  - Allows **network resource optimization** at the level of routing (eg: constrained based routing)
  - Allow **survivability and fast-reroute** features…
  - Can be **generalized** for optical networks (G-MPLS)