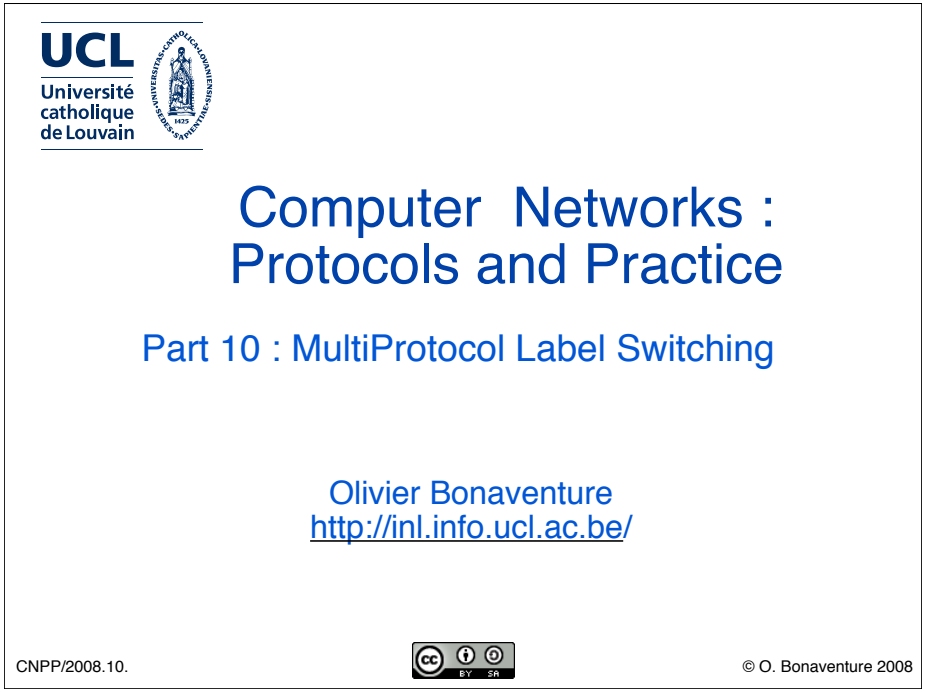


MPLS
MultiProtocol Label Switching

- Outline
  - Multiprotocol Label Switching
    - The label swapping forwarding paradigm
    - Integrating label swapping and IP
  - Utilisations of MPLS
    - Destination based packet forwarding
    - Simpler ISP backbones
    - Traffic engineering
    - QoS support
    - Fast restoration

CNPP/2008.10.

A good textbook on MPLS is the following

B.Davie and Y.Rekhter. *MPLS Technology and Applications*. Morgan Kauffmann, 2000.

A more practical book on MPLS, centered around Cisco routers is :
I. Pepelnjak and J. Guichard, MPLS and VPN Architectures, Cisco Press, 2001

The MPLS technology is standardised notably within IETF, see
http://www.ietf.org/html.charters/mpls-charter.html

Most of the standardisation documents on MPLS, including the deprecated ones may be found at
N.Demizu. Multi layer routing. Available from http://www.watersprings.org/links/mlr/.

High performance packet switching

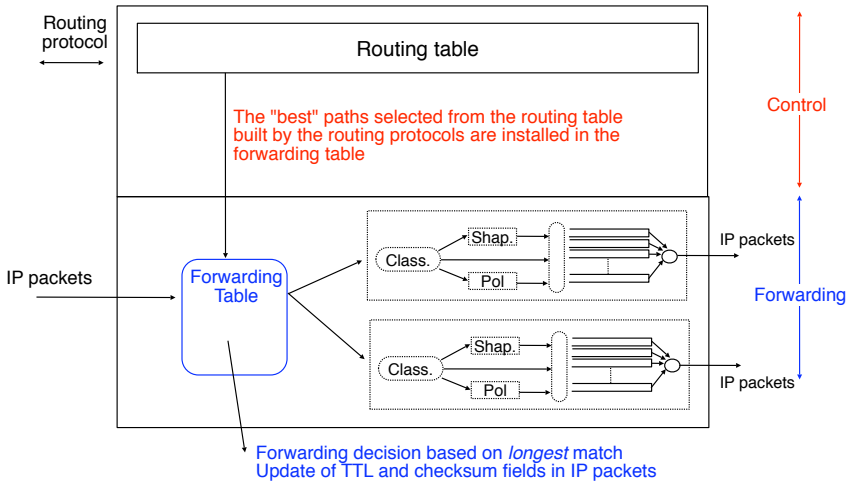
- The challenge
  - To efficiently support high speed networks, a router should be capable of switching and forwarding packets at line rate on all interfaces
  - Inter-packet time on high-speed interfaces

Inter-packet time in microsec versus packet size in bytes

Interface	40 bytes	250 bytes	1500 bytes
10 Mbps	32	200	1200
100 Mbps	3,2	20	120
155 Mbps	2,06	12,9	77,42
622 Mbps	0,51	3,22	19,29
2.4 Gbps	0,13	0,83	5
10 Gbps	0,03	0,2	1,2

  - Memory access time : 10 nsec for SRAM

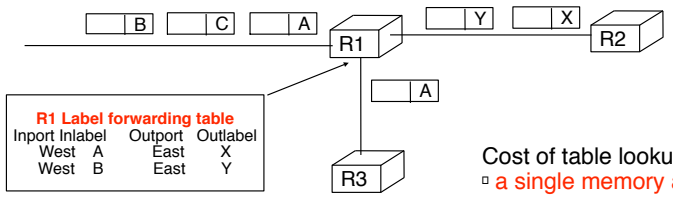
Architecture of a normal IP router



CNPP/2008.10.

Label swapping
A simpler forwarding paradigm

- Principle
  - On packet arrival, router analyzes
    - Packet Label
    - Input Port
  - Based on label forwarding label, router decides
    - Output Port for outgoing packet
    - Packet Label for outgoing packet



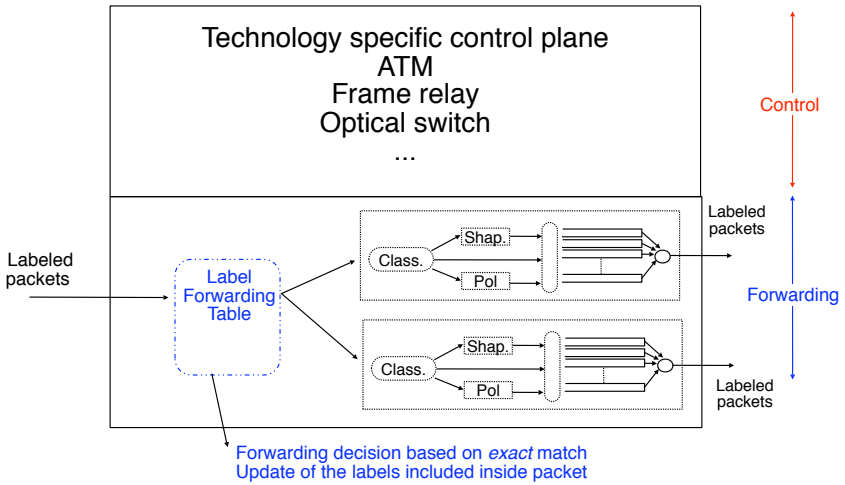
R1 Label forwarding table

Inport	Inlabel	Outport	Outlabel
West	A	East	X
West	B	East	Y

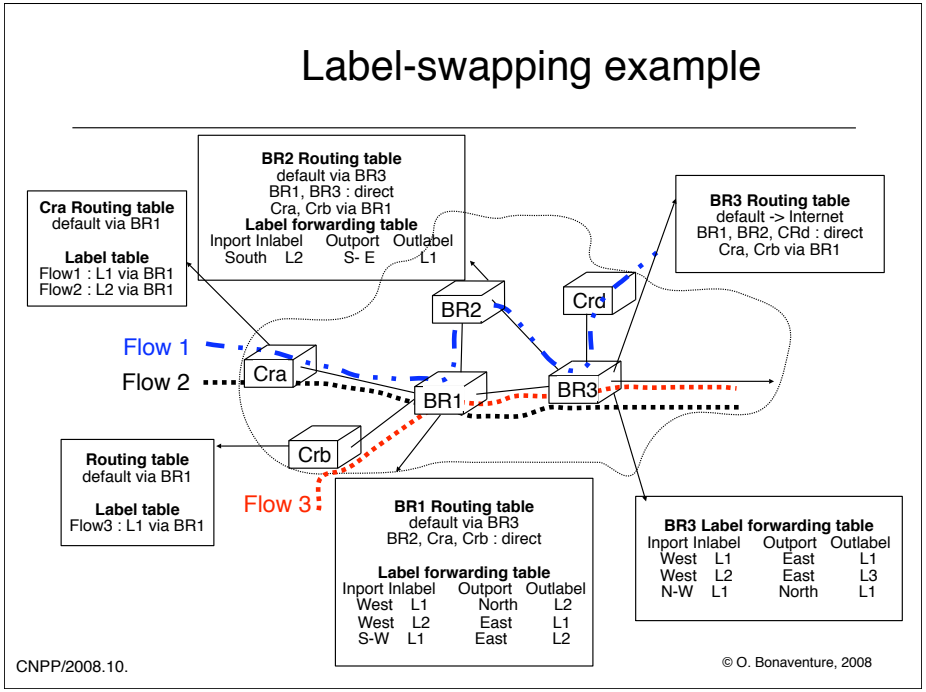
Cost of table lookup
- a single memory access !

Label swapping is notably used in ATM and Frame Relay networks.

Architecture of a label switch



CNPP/2008.10.



In principle, a MPLS flow can be considered as a layer 2.5 flow, namely a flow that belongs to an intermediate layer between layer 2 and layer 3.

Historical perspective

- Early 1990's
  - IP routers were essentially high-speed computers with special software
    - Impossible to implement IP forwarding in hardware
    - Performance of routers was limited by their CPU
      - forwarding assistance with caches to avoid expensive route lookups
  - Asynchronous Transfer Mode
    - Emerging technology relying on label swapping
    - Hardware implementation was easy at high speeds
      - 155 Mbps, 622 Mbps
    - To support voice (telephony) all packets were divided in 48 bytes cells with 5 bytes header for label

ATM is still widely used as the backbone for ADSL access, but many ADSL deployments are moving towards Ethernet-based access networks to replace ATM

A description of ATM may be found in :

Martin De Prycker, Asynchronous Transfer Mode. Solutions for Broadband ISDN (Prentice-Hall, 1993)

Other pointers may be found : http://en.wikipedia.org/wiki/Asynchronous_Transfer_Mode

Historical perspective (2)

- Early 2000's
  - Moore's Law
    - CPU performance doubles every 18 months
    - VLSI integration allows more complex CPUs
  - IP routers
    - New advances in forwarding algorithms
      - IP forwarding can be easily performed in hardware
      - chips are complex but wire speed at 2.4 Gbps or 10 Gbps works
    - Label swapping is not anymore necessary from performance or hardware implementation point of view
  - ATM
    - 48+5 bytes cell size is major drawback and implementation cost
    - ATM interfaces on IP routers do not exist above 622 Mbps
    - ATM mainly restricted to medium speed ADSL->1Gbps

Additional information about Gordon Moore's law may be found at http://www.intel.com/technology/mooreslaw/index.htm

Moore's law mainly applies to high volume product such as PC CPU's, it does not completely apply to the specialised ASICs found in routers, however, the

performance improvement for ASCIs is similar.

Historical perspective (3)

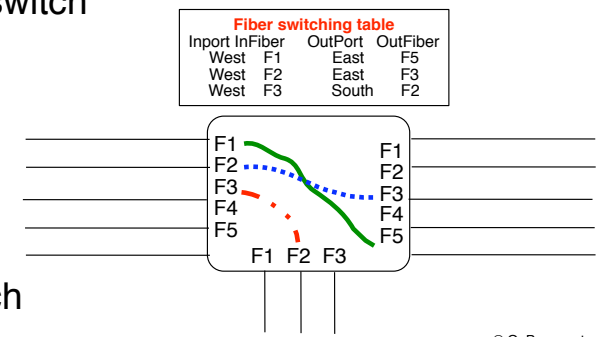
- Current motivations for MPLS
  - Applications
    - destination based routing
    - traffic engineering
    - QoS
    - fast restoration
    - Virtual Private Networks
  - Utilisation of MPLS to control optical or transmission devices close to label swapping
    - fibre switch
    - lambda switch
    - TDM (SONET/SDH) switches

A detailed description of the utilisation of MPLS to control optical networks is outside the scope of this course. A good description of this utilisation of MPLS

may be found in GMPLS: Architecture and Applications (The Morgan Kaufmann Series in Networking)by Adrian Farrel and Igor Bryskin

Historical perspective (4)

- MPLS controlled devices
  - MPLS and IP routing are used to establish flows through these devices
  - equipment uses a special forwarding mechanism
- Fibre switch

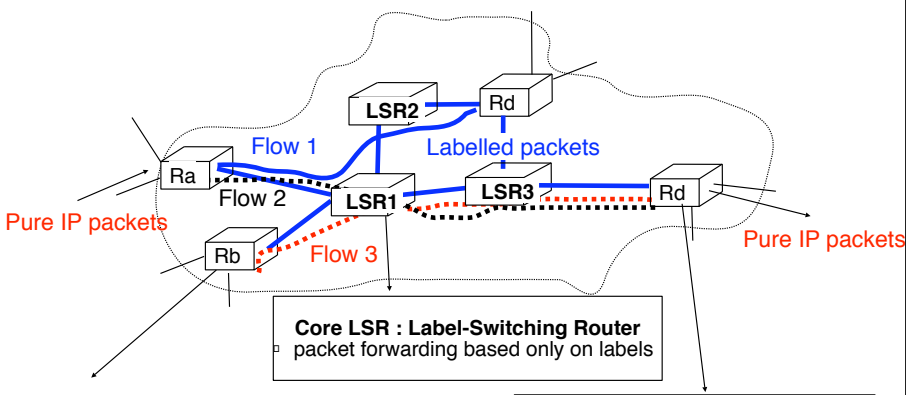


- λ switch

MPLS
MultiProtocol Label Switching

- Plan
  - Multiprotocol Label Switching
    - The label swapping forwarding paradigm
    - Integrating label swapping and IP
  - Utilisation of MPLS
    - Destination based packet forwarding
    - Simpler ISP backbones
    - Traffic engineering
    - QoS support
    - Fast restoration

Integrating label swapping and IP



Ingress Edge Router
- inserts labels on packets sent through backbone

Egress Edge router
- removes labels before sending packets outside MPLS network

CNPP/2008.10.

The main standardisation documents on MPLS may be found at http://www.ietf.org/html.charters/mpls-charter.html :

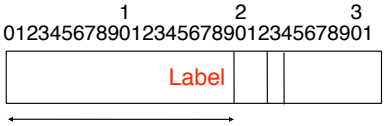
- R.Callon, E.Rosen, and A.Viswanathan. Multiprotocol label switching architecture. Internet RFC 3031, January 2001.

Integrating label swapping and IP (2)

- Problems to be solved
  - What is a labelled IP packet ?
  - What is the behaviour of a core LSR ?
  - What is the behaviour of an edge LSR ?

Labelled IP packets

- Generic solution
  - Insert special 32 bits header in front of IP packet



20 bits :1048576 different labels

- Technology specific solutions
  - Reuse the already available "labels" below layer 3
    - Frame Relay
    - Asynchronous Transfer Mode
    - Fibre/lambda switching with special label semantics
    - SONET/SDH with special label semantics

The encoding of the MPLS label is defined in :
E.Rosen, Y.Rekhter, D.Tappan, D.Farinacci, G.Fedorkow, T.Li, and A.Conta. MPLS label stack encoding. RFC 3032, 2001.

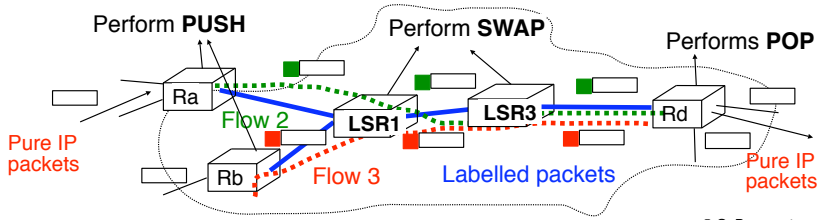
The utilisation of MPLS to support ATM and frame relay switches are discussed in :

B.Davie, J.Lawrence, K.McCloghrie, E.Rosen, G.Swallow, Y.Rekhter, and P.Doolan. MPLS using LDP and ATM VC switching. Internet RFC 3035, January 2001.

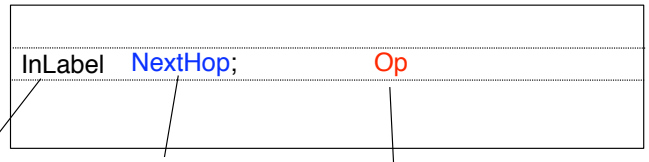
A.Conta, P.Doolan, and A.Malis. Use of label switching on frame relay networks specification. Internet RFC 3034, January 2001.

Operations performed on labelled packet

- Three types of operations
  - PUSH
    - insert a label in front of a received packet
  - SWAP
    - change the value of the label of a received labelled packet
  - POP
    - remove the label in front of a received labelled packet



Content of the Label forwarding table



- Identification of the label of the incoming packet
- Next hop for the packet
  - outgoing interface
    - packet sent to another LSR
  - LSR itself
    - packet destination is LSR
    - packet needs to be routed as a normal IP packet after pop
- Operation to be performed on label
  - swap label in front of packet
  - push new label in front of packet
  - pop label in front of packet

Label spaces

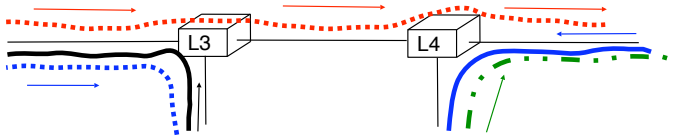
- A LSR may manage its labels with 2 methods
  - Per interface label space
    - one label space for each interface
    - 2^{20} distinct labels for each interface of the LSR

L3's per-interface Label forwarding table

Inlabel	Outport	Operation
West:L3	South	SWAP(L0)
West:L4	East	SWAP(L3)
South:L3	West	SWAP(L5)

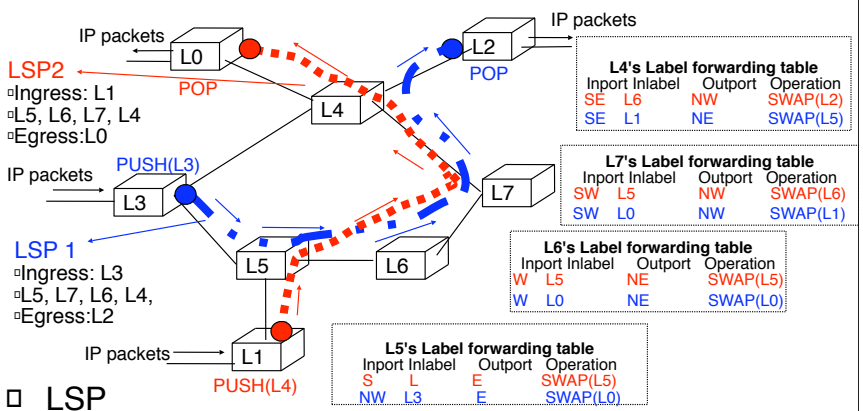
L4's per-LSR Label forwarding table

Inlabel	Outport	Operation
L0	East	SWAP(L0)
L4	South	SWAP(L3)
L3	East	SWAP(L5)



  - Per LSR label space
    - a single label space for all interfaces
    - 2^{20} distinct labels for the LSR

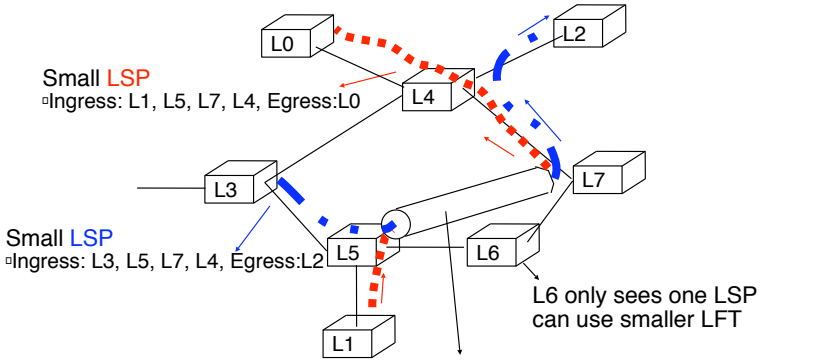
LSP
Label switched path



- LSP
  - a path followed by a labelled packet over several hops starting at an ingress LSR and ending at an egress LSR
  - a LSP is usually unidirectional (ingress -> egress)

In this case, we assume that per interface label space is used. The same example can of course be drawn for per-LSR label space.

How to improve scalability ?



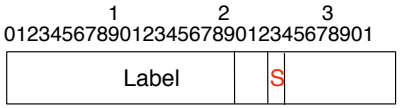
- Scalability of LSPs
  - it should be possible to place small LSPs inside large LSPs

CNPP/2008.10.

The need to support flows carrying other flows is common to most networking technologies. ATM networks rely on the utilisation of virtual paths that can carry a large number of virtual circuits.

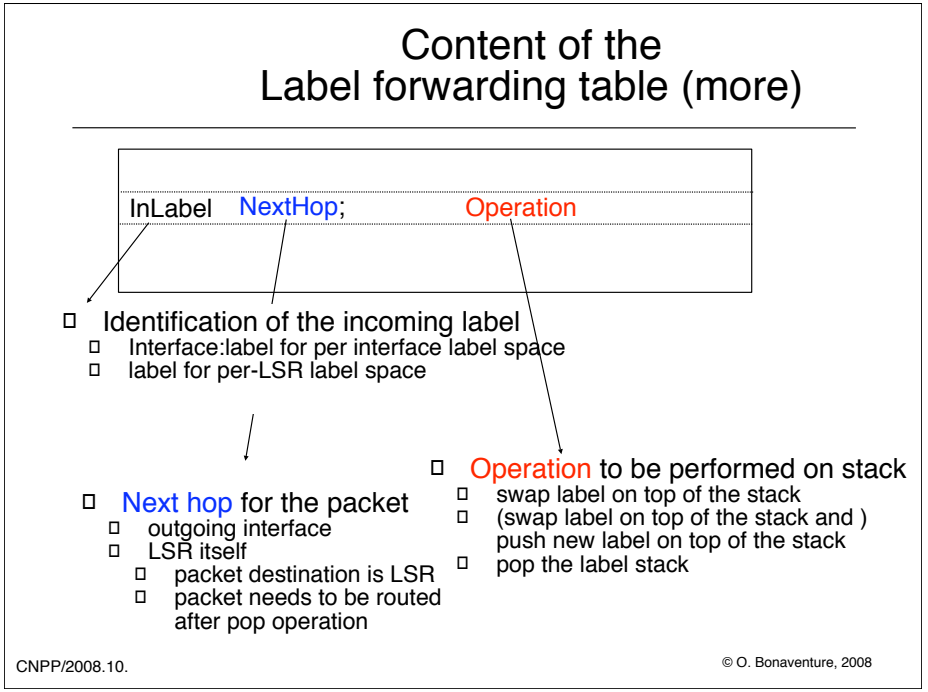
Labelled IP packets (more)

- How to support hierarchy of LSPs ?
  - it should be possible to place small LSPs inside large LSPs
  - ideally, there should be not predefined limit on the number of levels supported
- Solution adopted by MPLS
  - each labelled packet can carry a stack of labels



- label at the top of the stack appears first in packet
  - S=1 if the label is at the bottom of the stack
  - S=0 if the label is not at the bottom of the stack

The stack of labels is one of the major innovations of MPLS compared to the other label-based forwarding techniques. The utilisation of a stack is, of course, the reason why the two basic operations of ingress and egress LSRs are called push and pop.



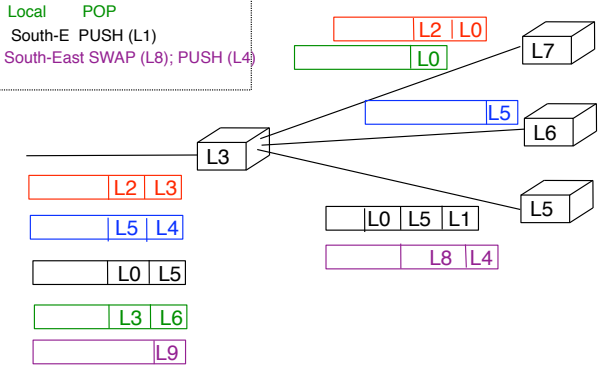
In the case of multicast LSPs it should be possible to establish LSPs as trees. In this case, a LSR may serve as a branch point and a packet might need to be replicated on several outgoing interfaces. For this reason, each line of the label forwarding table may contain several pairs (NextHop; Operation), one for each outgoing interface on the multicast tree.

Content of the Label forwarding table (more) (2)

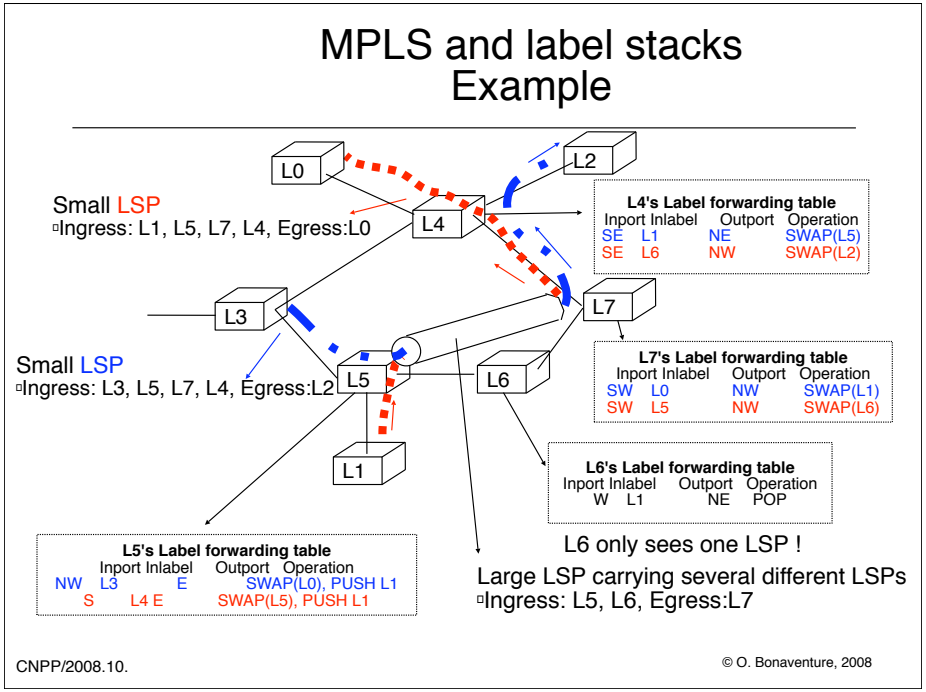
- Example

L3's Label forwarding table

Inlabel	Nexthop	Operation
L3	North-E	SWAP(L0)
L4	East	POP
L6	Local	POP
L5	South-E	PUSH (L1)
L9	South-East	SWAP (L8); PUSH (L4)



In this example, the packet received with label L6 has a special treatment. First, label L6 is popped and the destination is the local LSR. After the POP operation, the packet contains label L3 and according to the first line of the label forwarding table, this label is swapped to L0 and the packet is sent on the NE link.



In practice, the large LSP between L5 and L7 would pass through several intermediate LSRs. For graphical reasons, only L6 is shown on the slide.

Behaviour of ingress edge LSR

- How does edge LSR at ingress determine the label to be used to forward a received packet ?
  - Principle
    1. Divide the set of all possible packets into several Forwarding Equivalence Classes (FEC)
       - *a FEC is a group of IP packets that are forwarded in the same manner (e.g. over the same path, with the same forwarding treatment)*
       - examples
         - all packets sent to the same destination prefix
         - all packets sent to the same BGP next hop
    2. Associate the same label to all the packets that belong to the same FEC

The FEC is defined in RFC3031

Behaviour of ingress edge LSR (2)

- Example

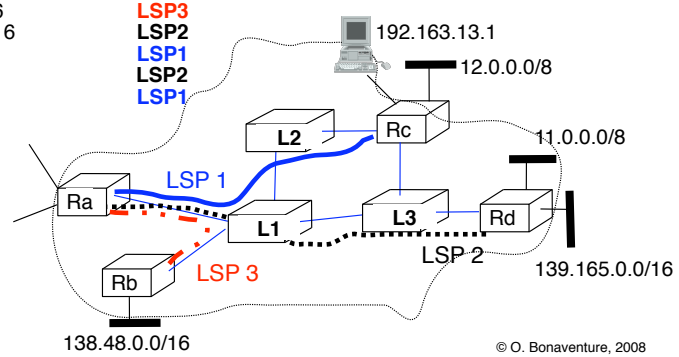
LSP1
- Ingress=Ra, L1, L2, Egress=Rc

LSP2
- Ingress=Ra, L1, L3, Egress=Rd

LSP3
- Ingress=Ra, L1, Egress=Rb

Ra's Mapping table

IP dest prefix	LSP
138.48.0.0/16	**LSP3**
139.165.0.0/16	**LSP2**
12.0.0.0/8	**LSP1**
11.0.0.0/8	**LSP2**
192.163.13.1	**LSP1**



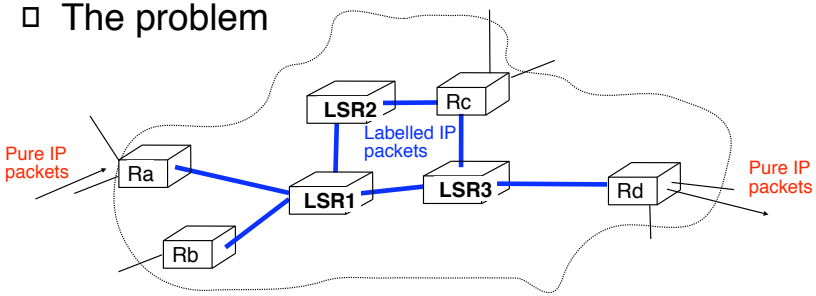
In this example, edge LSR Ra groups all the packets sent to the same egress router (Rb, Rc, Rd) inside a single LSP. Other mappings are of course possible, but this mapping the most frequently often used in practice.

MPLS
MultiProtocol Label Switching

- Plan
  - Multiprotocol Label Switching
    - The label swapping forwarding paradigm
    - Integrating label swapping and IP
  - Utilisations of MPLS
    - Destination based packet forwarding
    - Simpler ISP backbones
    - Traffic engineering
    - QoS support
    - Fast restoration

Destination-based packet forwarding

- The problem



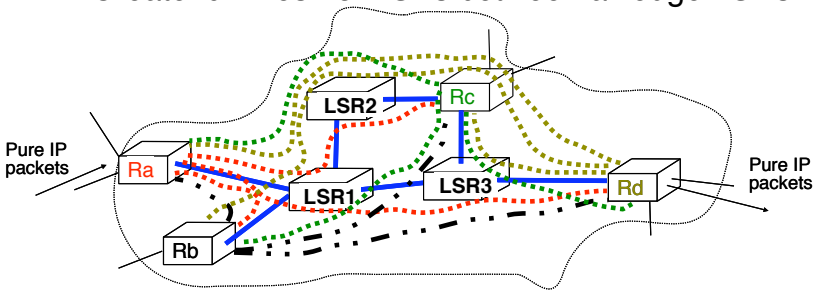
- How to provide transit service when
  - Edge LSRs are able to attach and remove labels
  - Edge LSR and Core LSRs run IP routing protocols and maintain IP routing tables
  - Core LSR can *only* forward labelled packets
    - Core LSR cannot route IP packets efficiently !

CNPP/2008.10.

The destination-based packet forwarding problem was initially solved to allow LSRs that were not able efficiently forward IP packet at high performance. It is now being used only with high-end routers/LSRs in the backbone of large networks. Those routers are able to forward IP packets and labelled packets at line rate, but the benefits of MPLS imply that MPLS is still used even if this is not for pure performance reasons.

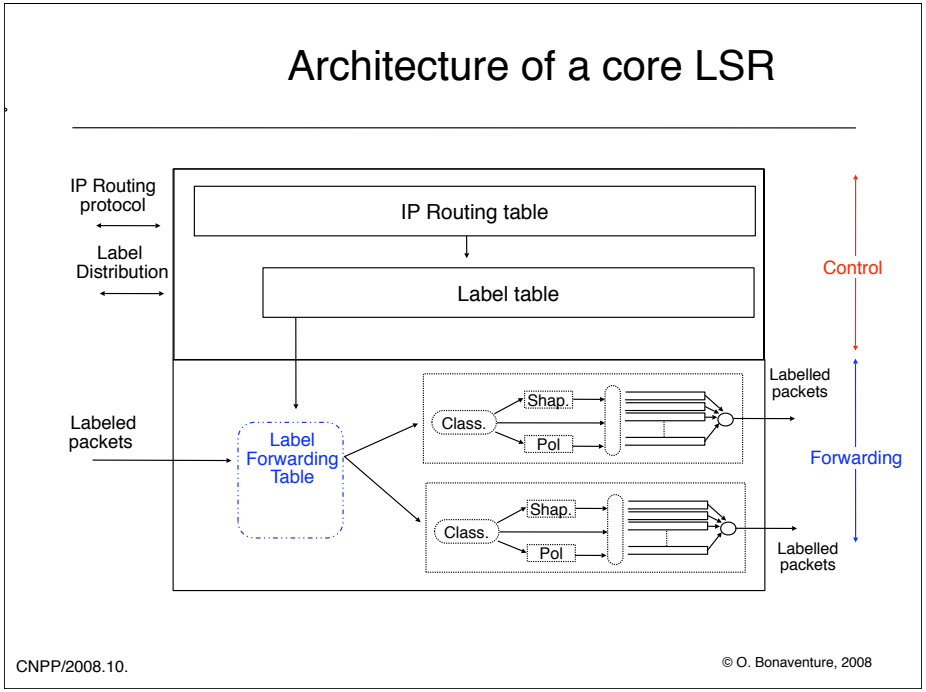
Destination-based packet forwarding (2)

- Manual solution
  - Create full mesh of LSPs between all edge LSRs



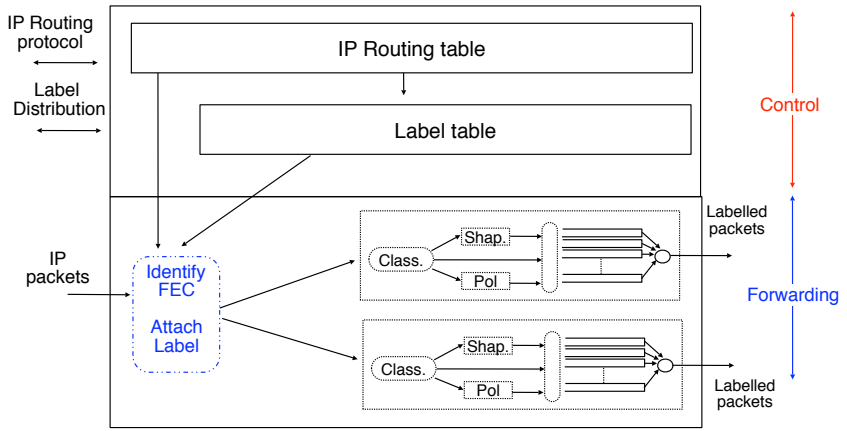
- Problems to be solved
  - N edge LSRs -> N*(N-1) unidirectional LSPs
    - How to automate LSP establishment ?
    - How to reduce the number of required LSPs ?

The main issue with the full mesh of LSPs is the size of the label forwarding tables in the core LSRs. If each LSR needs to maintain a table with N^2 entries, this may create performance or memory problems in large networks. Although MPLS supports 2^{20} different labels on each interface, MPLS allows LSRs to only support a limited number of labels. This number could depend on the amount of memory available on the LSR or on its interfaces.



In practice, a core LSR is also able to forward and route IP packets, but the achieved performance is often much lower than with labelled packets. This is the reason why IP packet forwarding is not shown in this architecture.

Architecture of an ingress edge LSR



CNPP/2008.10.

This figure corresponds to the ingress part of an edge LSR.
The egress part of an edge LSR will receive labelled IP packets and will remove the labels before acting as a normal IP router.

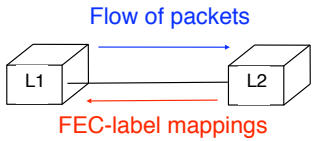
Distributing labels

- How to fill the label forwarding tables of all LSRs in a given network ?
  - Use a special protocol to distribute FEC-label mappings
    - LDP : Label Distribution Protocol
    - RSVP-TE : extensions to RSVP
  - Piggyback FEC-label mappings inside messages sent by routing protocol
    - possible if routing protocol is extensible
      - BGP can be easily modified to associate label with route
      - RIP cannot be used because its syntax is not extensible
      - link-state protocols (OSPF IS-IS) do not distribute routes

We discuss in this section the LDP protocol. The utilisation of RSVP and BGP to distribute labels will be explained later.

How to distribute labels ?

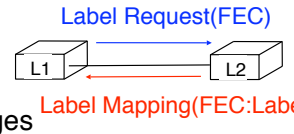
- Who determines the FEC-label mapping ?
  - packets are sent by upstream LSR towards downstream LSR
  - FEC-label mappings are sent by downstream LSR towards upstream LSR



This way of allocating the labels is useful for implementation reasons. For example, consider LSR that uses high-speed label forwarding tables that contain only 1024 entries. With a downstream allocation of the labels, this LSR can simply only allocate the values between 0 and 1023 and will be sure to receive packets with those values. The lookup in the label forwarding table can be implemented as a direct access to LFT[L] where L is the received label.
If the upstream node was allocating the labels, then the LSR should be able to receive packets with any label values. To use a LFT with 1024 entries, it could rely on a secondary table used to map received labels to LFT indexes. The LFT lookup would then be implemented as LFT[index[L]] at the cost of two memory access.
An alternative would be to use hash functions instead of a table, but then the hash collisions must be taken into account.

Label distribution modes

- When to distribute the FEC-label mapping?
  - Downstream on demand label distribution
    - upstream LSR requests and downstream allocates label

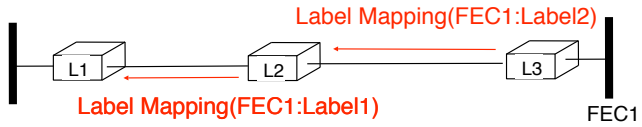


    - advantages
      - the FEC-label mappings are only distributed when needed
      - LSR only need to store the FEC-label mappings that are in use
    - drawback
      - when a next-hop fails, some time may elapse while a new FEC-label mapping is been requested from the new next-hop

We will see RSVP-TE later as an example of downstream on deman label distribution.

Label distribution modes (2)

- When to distribute the FEC-label mapping ?
  - Unsolicited downstream label distribution
    - downstream LSR announces independently FEC-label mappings to upstream LSR



    - advantage
      - Each LSR can obtain several labels for each FEC
      - in case of failure, LSR can quickly switch from one label to another
    - drawback
      - Labels may not be distributed at the best time

We will see LDP and BGP as examples of unsollicited downstream label distributions.

Label Distribution Protocol

- LDP : Label Distribution Protocol
  - Designed to distribute FEC-labels mapping on a hop-by-hop basis inside network when labels cannot be distributed by routing protocols
  - Neighbour discovery over UDP
    - determine whether the neighbour is a LSR or a normal router
  - Distribution of FEC-label mappings over TCP
    - several modes of distribution are supported by LDP
    - we will only provide some examples of LDP
    - Main messages
      - Initialisation
        - establishment of a LDP session
      - Keepalive
        - used to verify that the LDP session is still up
      - Label mapping
        - used by LSR to announce a FEC-label mapping
      - Label withdrawal
        - Used by LSR to withdraw a previous FEC-label mapping

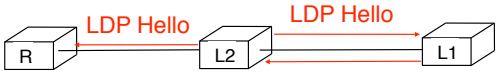
CNPP/2 08

The Label Distribution Protocol is defined in :
L.Andersson, P.Doolan, N.Feldman, A.Fredette, and B.Thomas. LDP specification. Internet RFC 3036, January 2001.
B. Thomas, E. Gray, LDP Applicability, RFC 3037, 2001

Additional details may be found in :
C.Boscher, P.Cheval, L.Wu, and E.Gray. LDP state machine. RFC 3215, January 2002.

Neighbour discovery

- Principle
  - LSR periodically send LDP Hello packets to neighbor on "all routers" multicast address
    - LDP neighbour discovery uses UDP port 646
  - neighbours respond with LDP Hello if they are LSR



L1 is LSR
R is not LSR

- LSR with highest IP address becomes active and establishes TCP connection for LDP on port 646
- LSR with lowest IP address becomes passive and waits the establishment of TCP connection for LDP session
  - TCP session established on port 646
  - LDP session establishment allows negotiation of options

The initialisation of the LDP session is done by sending the INITIALIZATION message. This message may contain several options. If the remote LSR accepts the LDP session with the proposed options, it replies with a KEEPALIVE message.

KEEPALIVE messages are regularly exchanged over each LDP session to ensure that the LDP session is still up and running and to detect failures.

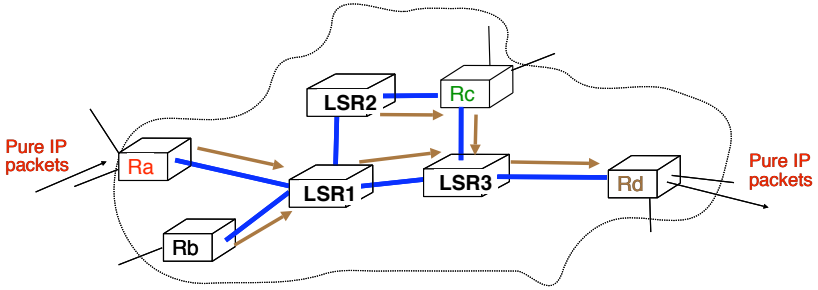
LDP messages

- Initialization
  - used during LDP session establishment to announce and negotiate options
- Keepalive
  - sent periodically in absence of other messages
- Label mapping
  - used by LSR to announce a FEC-label mapping
- Label withdrawal
  - used by LSR to withdraw a previous FEC-label mapping
- Label Release
  - used by LSR to indicate that it will not use a previously received FEC-label mapping
- Label Request
  - used by LSR to request a label for a specific FEC

We do not cover the label-release and label request messages in this presentation.

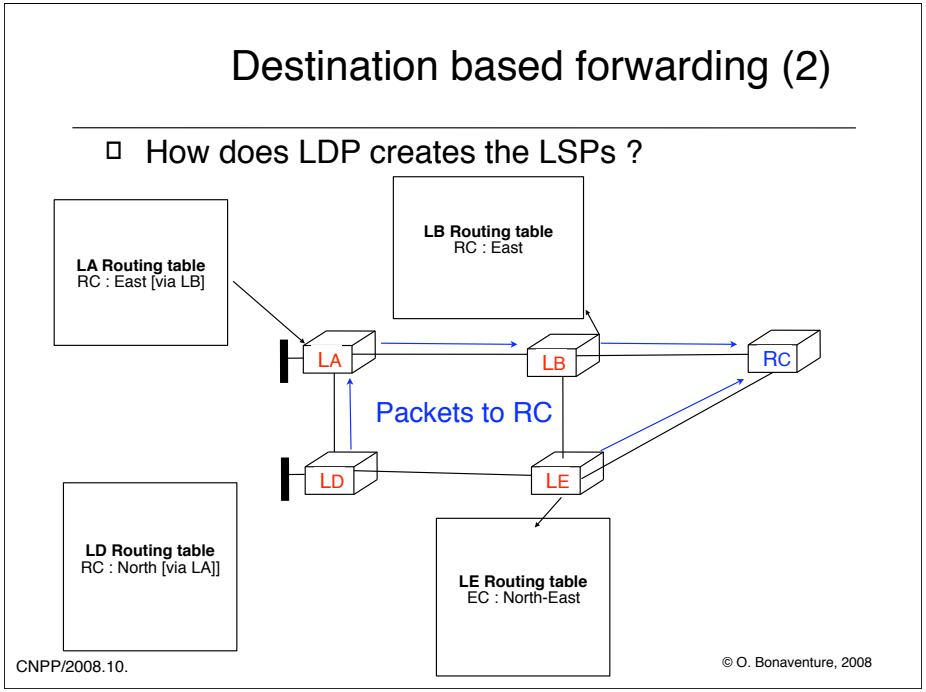
Destination based forwarding

- Principle
  - Labelled packets should follow the same path through the network as if they were pure IP packets



- create a tree shaped LSP rooted on each egress LSR
  - similar to the way IP routing would forward packets
  - one tree per egress LSR, reduces total number of LSPs
- distribute the labels to build those trees

Note that by using a tree-shaped LSP, it is possible to significantly reduce the size of the label forwarding tables of the core LSR.

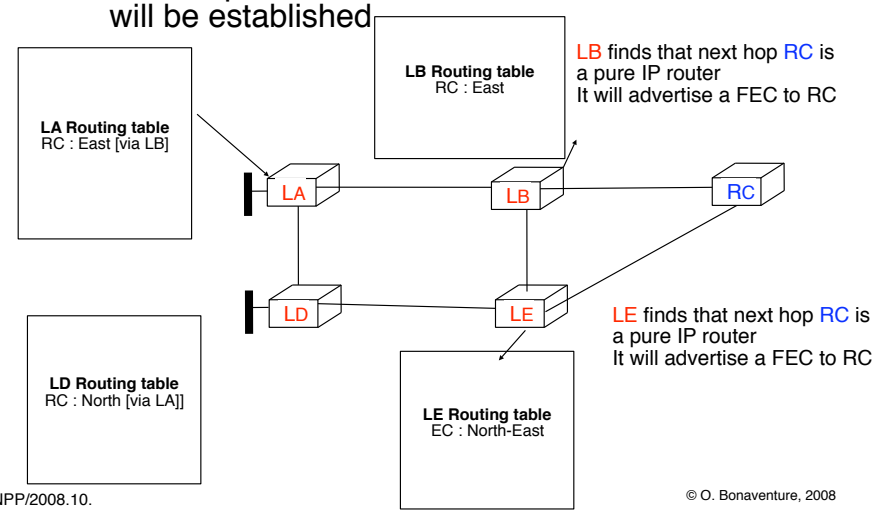


In this example, we assume that the link between LD and LE is a lower bandwidth link whose metric is higher than the other links in the network. For this reason, the link is not preferred by the IGP. The other links have the same metric.

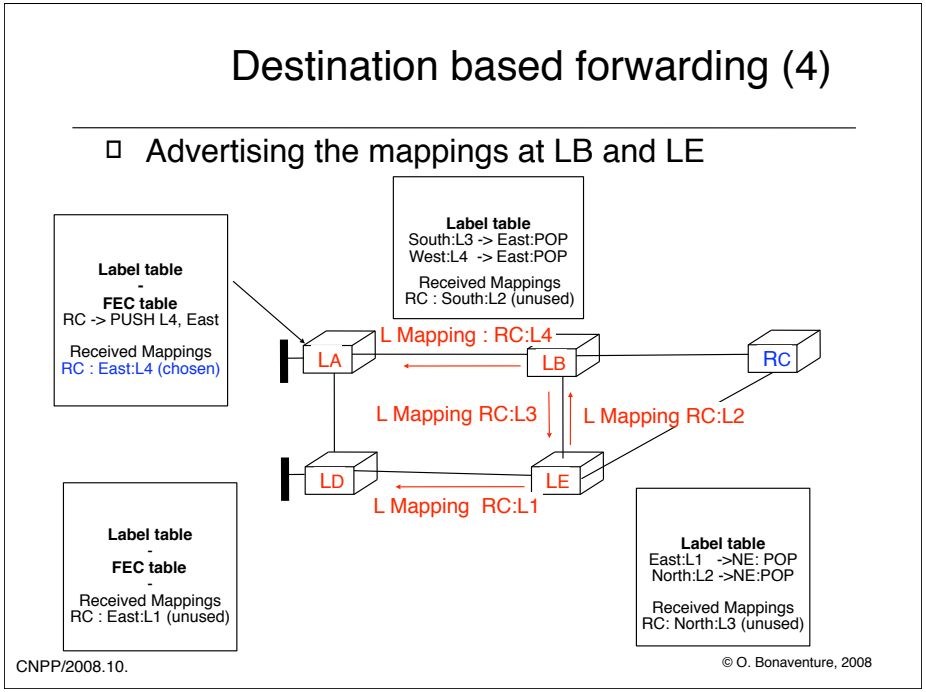
To be able to create an LSP, LDP needs to know the routes for each destination for which a LSP will be established. In practice, the routing table will be much larger than as shown on the slide.

Destination based forwarding (3)

- First step : choose destination for which a LSP will be established



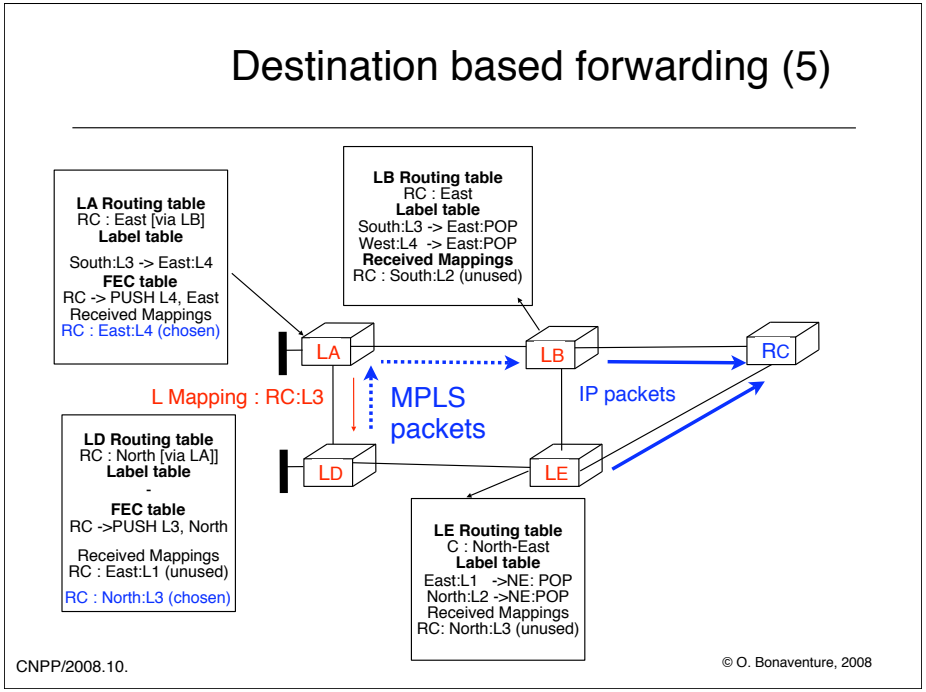
In the following slides, we will not show anymore the IP routing tables, but they are used by LDP



At this point, LE is able to receive MPLS packets from LD and LB. However, since the IGP path to RC from LB and LD is not via LE, LE will not receive such MPLS packets.

The first LSP that has been established is the one between LA and LB. LA has chosen the mapping advertised by LB because its IGP path to reach RC is via LB.

LE did not select the mapping advertised by LB. LB did not select the mapping advertised by LE.

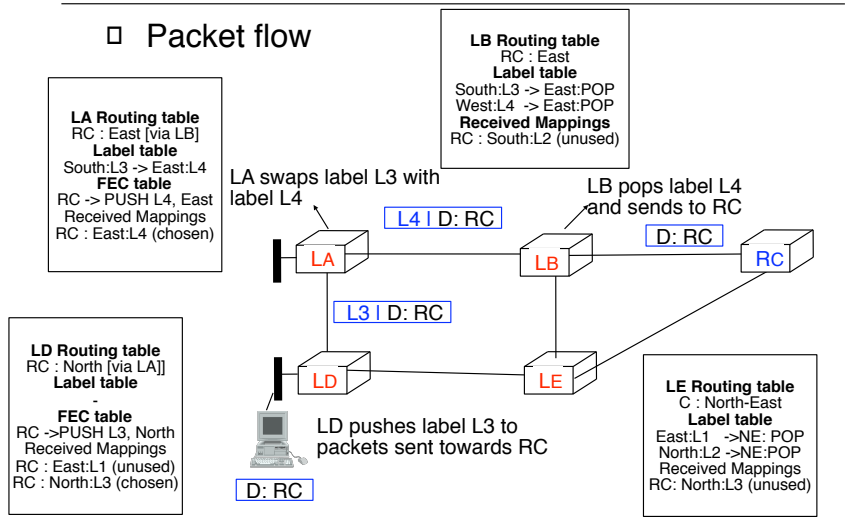


The blue arrows show the tree-shaped LSP used by LD and LA to send labelled packets whose destination is RC via LB which is their preferred egress LSR. LE sends directly its packets to RC.

Note that the MPLS packets flow in the opposite direction of the mappings.

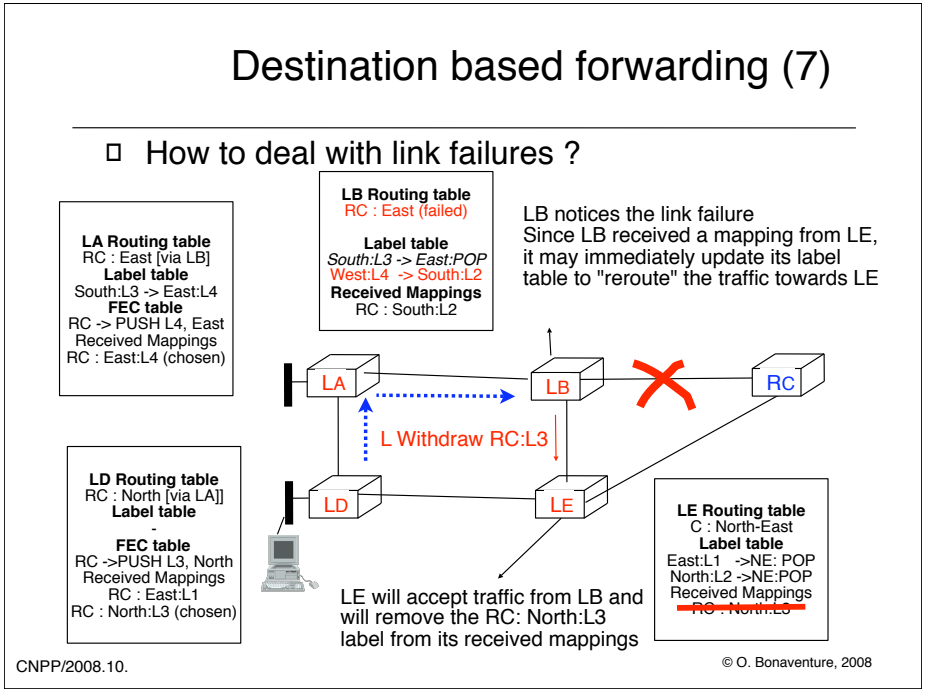
Destination based forwarding (6)

- Packet flow



CNPP/2008.10.

This example shows the transmission of one packet from the workstation attached to LD to destination RC.

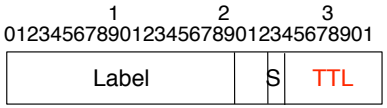


When LSR LB detects the fail of the link that it uses to reach RC, it can immediately update its label forwarding table to use the label that it received from LE. LB does not need to change the label that it advertised to LA since it is still able to reach RC via LE. However, LB must inform LE that the label L3 that it advertised earlier is not anymore available. This is done with the label withdraw message.

After some time, the routing protocol will update the routing tables to reflect the link failure. The updates to the routing tables will trigger the distribution of new label mappings.

MPLS and transient loops

- What if routing has created a transient loop while LSP is being established ?
  - LSP could be looped inside network
- How to recover from looped LSPs ?
  - Discard looped packets as in IP
    - TTL field in MPLS header

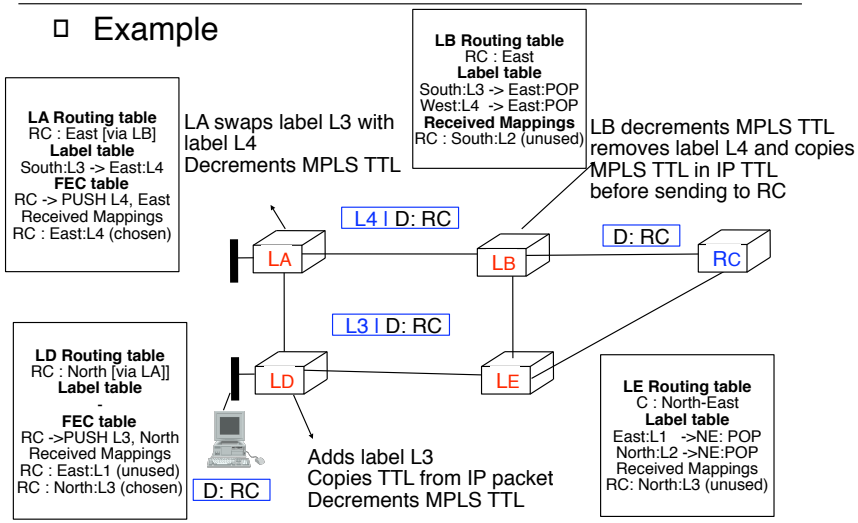


    - not suitable for technologies with their own header (ATM)
  - Prevent loops inside LDP
    - indicate the entire path of LSP in LDP Label request and LDP label mapping messages

We do not discuss in this presentation the utilisation of LDP to prevent loops.

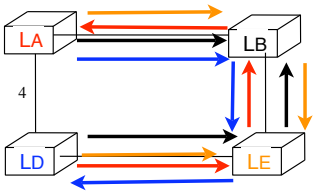
Utilisation of the MPLS TTL

- Example



LDP in practice

- How do network operators use LDP ?
- Most common deployment is to create a full mesh of LSPs among all LSRs so that each LSR is able to send MPLS packets to any LSR
  - LSR usually advertises a label for its loopback
  - What will be the LSPs in the network below ?



CNPP/2008.10.

Using loopback addresses to advertise mappings is preferred over addresses associated to a physical interface because a loopback address is always up and does not become unreachable if a physical interface fails

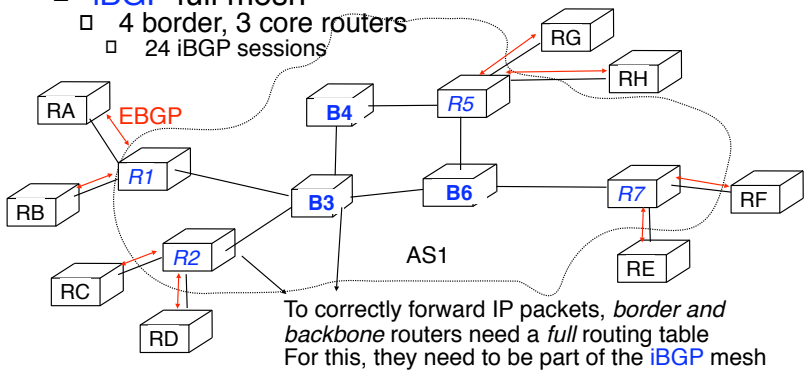
In the example network above, the metric of all links is set to 1 except the link between LA and LD.

MPLS
MultiProtocol Label Switching

- Outline
  - Multiprotocol Label Switching
    - The label swapping forwarding paradigm
    - Integrating label swapping and IP
  - Utilisation of MPLS
    - Destination based packet forwarding
    - Simpler ISP backbones
    - Traffic engineering
    - QoS support
    - Fast restoration

MPLS in large ISP networks

- Pure IP-based ISP network
  - eBGP on border routers
    - current full BGP Internet routing table
      - +-220.000 active routes
  - iBGP full mesh
    - 4 border, 3 core routers
      - 24 iBGP sessions



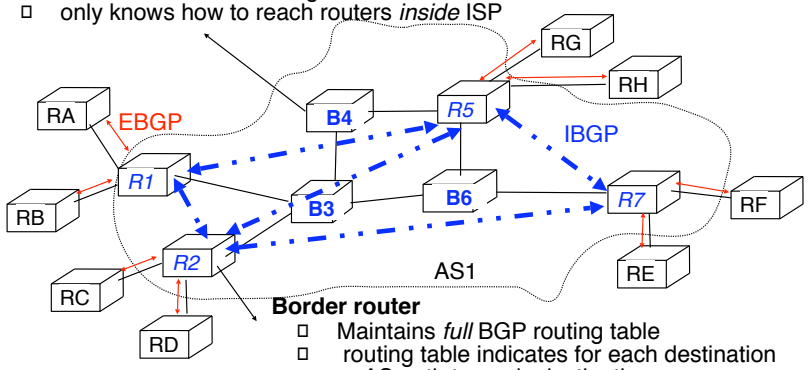
To correctly forward IP packets, *border and backbone* routers need a *full* routing table
For this, they need to be part of the iBGP mesh

MPLS in large ISP networks (2)

- BGP free ISP backbone

Backbone router

- Maintains *internal* routing table of ISP network
- only knows how to reach routers *inside* ISP



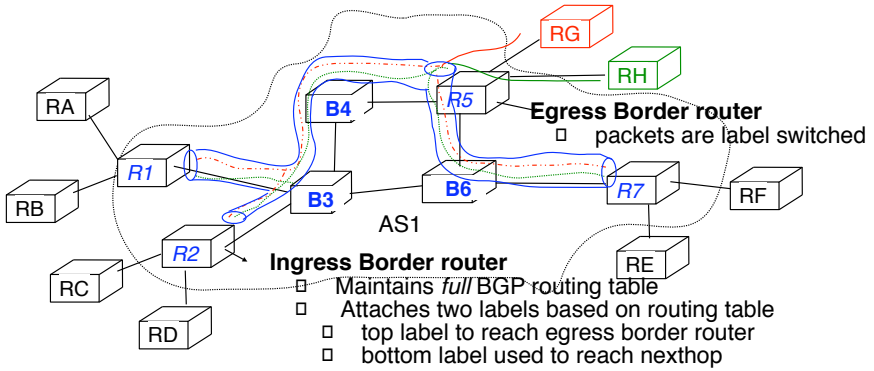
Border router

- Maintains *full* BGP routing table
- routing table indicates for each destination
  - AS path towards destination
  - IP address of next-hop to reach destination

In a large ISP network, there are two ways to advertise with BGP the nexthop used to reach a destination. For example, consider the prefix *p* advertised by router RG via eBGP to router R5.
The normal operation with BGP is that R5 advertises prefix *p* to its iBGP peers with RG as a nexthop. However, this solution requires the ISP to advertise in its IGP all its interdomain links (e.g. R5-RG, R5-RH, ...), which increases the size of the IGP tables.
A second solution that is often used is the allow the border router that receives a route via eBGP to advertise itself as the nexthop when advertising the route to iBGP peers. In the case of prefix *p*, router R5 would advertise its prefix with itself as the nexthop to its iBGP peers. This requires a special configuration on the BGP routers. We assume this special configuration is used for the example described in this section.

MPLS in large ISP networks (3)

- Principle of the solution
  - Use a hierarchy of labels
    - top label is used to reach egress border router (blue LSP)
    - second label is used to reach eBGP peer (red/green LSP)

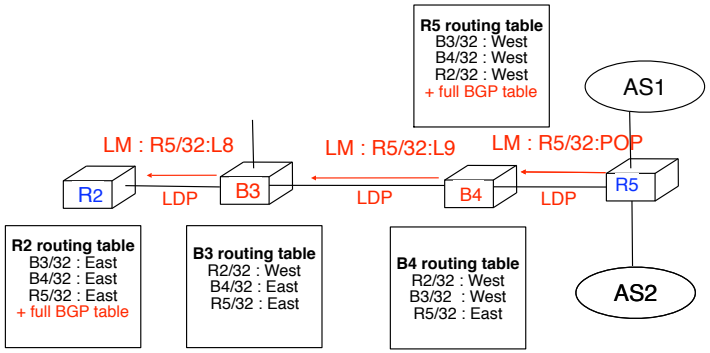


CNPP/2008.10.

This slide shows the LSPs that are used to reach RG and RH, the two eBGP peers of R5 from three ingress border routers : R1, R2 and R7. The comments associated to R2 and R5 show the operations performed when packets are sent from RC or RD towards RG or RH.

MPLS in large ISP networks (4)

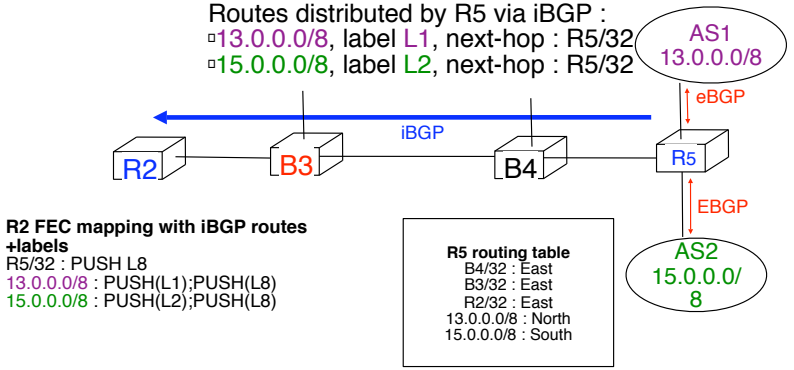
- How to distribute the two labels ?
  - LDP allows to distribute the label to reach the egress BGP router



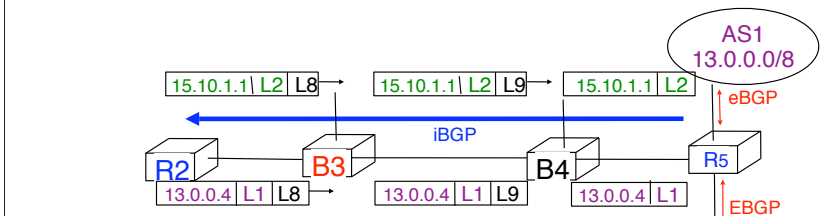
There is an iBGP session, not shown in the slide, between all border routers of the ISP, including R2 and R5. B3 and B4 do not participate in the iBGP full mesh and thus do not maintain BGP routing tables.

MPLS in large ISP networks (5)

- How to distribute the two labels ?
  - BGP allows to distribute the label associated to each prefix



MPLS in large ISP networks (6)



R2 FEC mapping with iBGP routes +labels
R2/32 : PUSH L8
13.0.0.0/8 : PUSH(L1);PUSH(L8)
15.0.0.0/8 : PUSH(L2);PUSH(L8)

B4 routing table
B3/32 : West
R2/32 : West
R5/32 : East

Label table
West:L9 -> East:POP

R2 routing table
B4/32 : East
B4/32 : East
R5/32 : East

LDP FEC mapping
R2/32 : PUSH L8

B4 routing table
R2/32 : West
B4/32 : East
R5/32 : East

Label table
West:L8 -> East:L9

R5 routing table
B4/32 : East
B3/32 : East
R2/32 : East
13.0.0.0/8 : North
15.0.0.0/8 : South

R5 Label table
West:L3 -> Local:POP
L1 -> North:POP
L2 -> South:POP

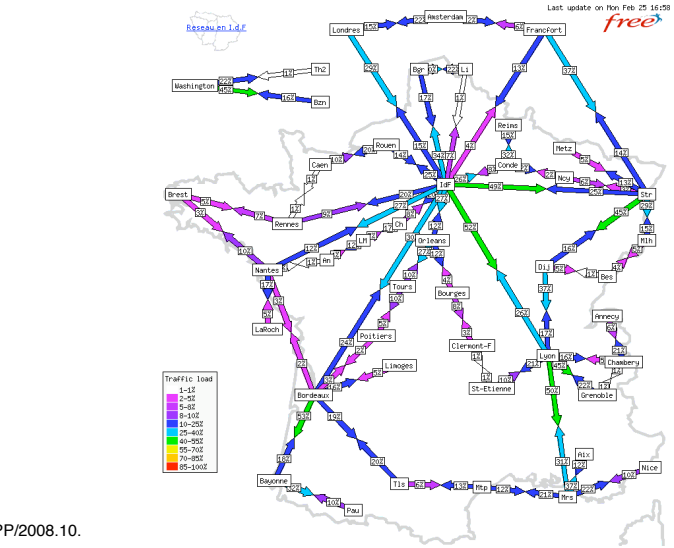
CNPP/2008.10.

The modifications to BGP to support the distribution of labels are described in :Y.Rekhter and E.Rosen. Carrying label information in BGP-4. Internet RFC 3107, May 2001.

MPLS
MultiProtocol Label Switching

- Outline
- Multiprotocol Label Switching
  - The label swapping forwarding paradigm
  - Integrating label swapping and IP
- Utilisation of MPLS
  - Destination based packet forwarding
  - Simpler ISP backbones
  - Traffic engineering
  - QoS support
  - Fast restoration

Traffic load in large IP networks



CNPP/2008.10.

Source : http://support.free.fr/reseau/carte_reseau.html built with http://netmon.grnet.gr/weathermap/

Other examples maybe found at

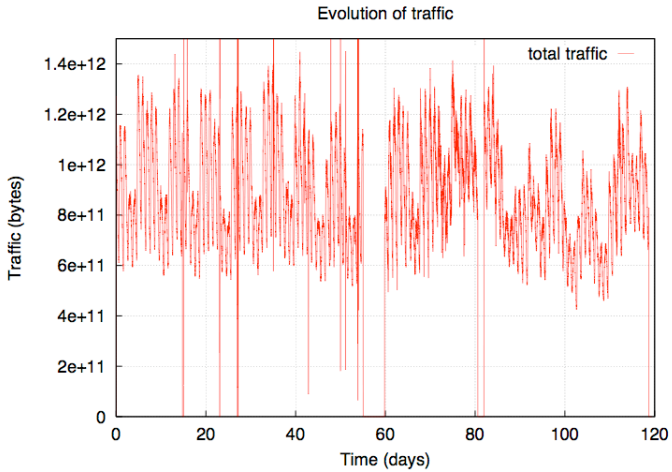
Internet2 weather map : http://atlas.grnoc.iu.edu/l2.html

nordunet
http://www.nordu.net/stat-q/load-map/ndn-map,,traffic,peak

http://monitor.belnet.be

Evolution of traffic load

- Example with GEANT



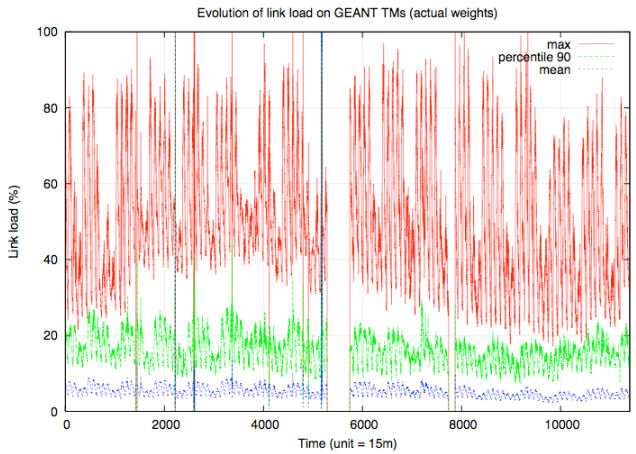
CNPP/2008.10.

The GEANT topology may be found at : www.geant.net/upload/pdf/Visio-AP01G009_SCH0102_2-GeantTopologya3(T)Visio2000Isis.pdf

The measurements were presented in S. Uhlig, B. Quoitin, S. Balon, and J. Lepropre. Providing public intradomain traffic matrices to the research community. ACM SIGCOMM Computer Communication Review, 36(1), http://inl.info.ucl.ac.be/publications/providing-public-intradomain-traffic-

Evolution of link load

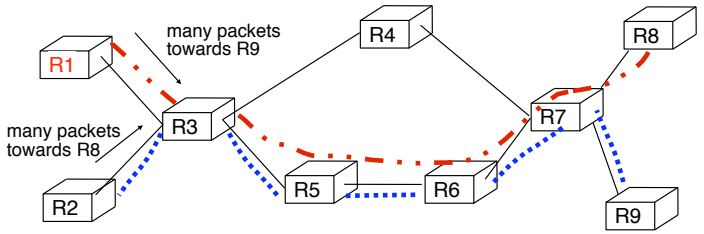
- Example with GEANT



CNPP/2008.10.

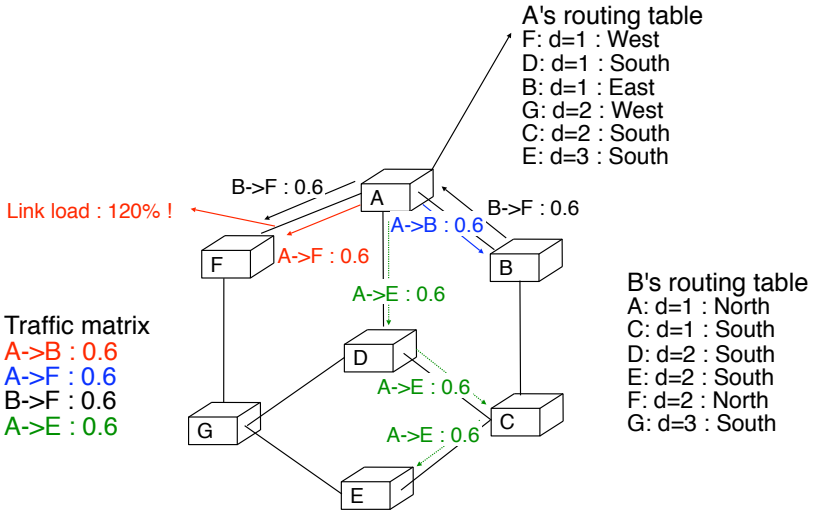
Traffic engineering

- Problem
  - Shortest path chosen by IP routing does not always lead to a good network utilisation
    - fish problem



  - How to better optimise the network utilisation ?
  - How to react to changes in traffic conditions

Traffic engineering simple case study



In this example, we assume that each link can carry one unit of traffic.

IP based traffic engineering

- How to solve the traffic engineering problem in a pure IP network ?
  - Two types of solutions

  - Router-level traffic engineering
    - Allow a router to use several paths instead of a single one for a given route
    - Possible with most router implementations

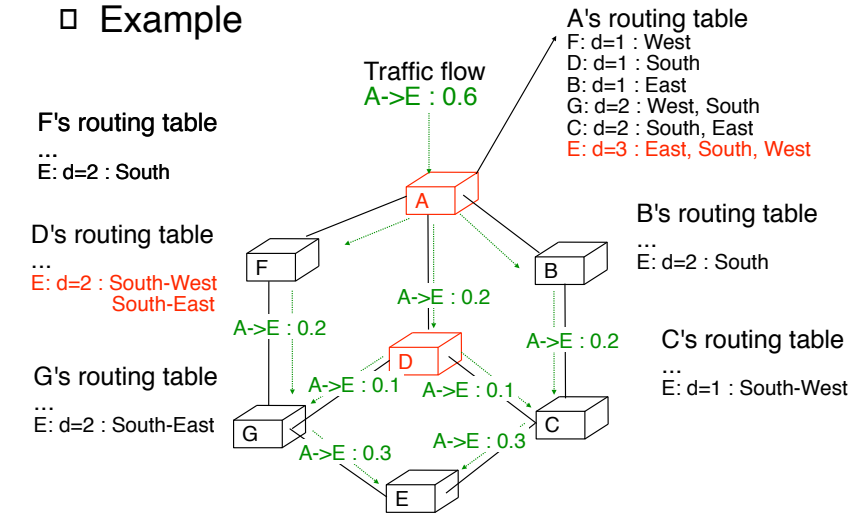
  - Network-level traffic engineering
    - Force aggregate traffic flows to follow some paths inside the network
    - Possible in some cases by playing with link costs

Equal Cost Multipath

- Simple network-level load balancing mechanism supported in OSPFv2
  - Principle
    - OSPF distributes the complete network topology to all routers inside network
    - based on this topology, each router computes the routes towards all destinations
    - if a router finds several equal cost paths reaching one destination, it may balance its traffic over these paths
    - load balancing is done at the discretion of this router without coordinnating with other routers
      - since routes are equal cost routes, loops will not occur provided that the routing table is stable

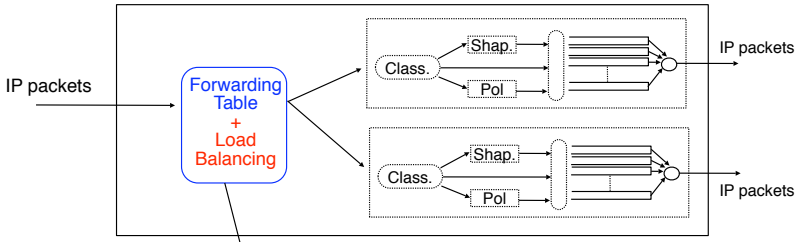
Equal Cost Multipath (2)

- Example



How to dispatch IP packets ?

- Principle
  - For each destination, remember the P equal paths instead of a single one
    - place those paths in forwarding table
  - when a packet arrives, load balancing algorithm selects one path among the P available paths



Forwarding decision based on *longest* match
Load balancing implies that forwarding table may contain several entries for each destination

Load balancing algorithms

- Simple solution
  - (Deficit) Round-Robin or variants to dispatch packets on a per packet basis
  - Advantages
    - easy to implement since number of paths is small
    - traffic will be divided over the equal cost paths on a per packet basis
      - each path will carry the same amount of packets/traffic
  - Drawbacks
    - two packets from the same TCP connection may be sent on different paths and thus be reordered
      - TCP performance can be affected by reordering

References to load balancing algorithms include :

C. Hopps, Analysis of an Equal Cost MultiPath algorithm, RFC2992, Nov. 2000

Z. Caro, Z. Wang, E. Zegura, Performance of Hashing-Based Schemes for Internet Load Balancing, INFOCOM2000, http://www.ieee-infocom.org/2000/papers/650.ps

Per TCP connection load balancing

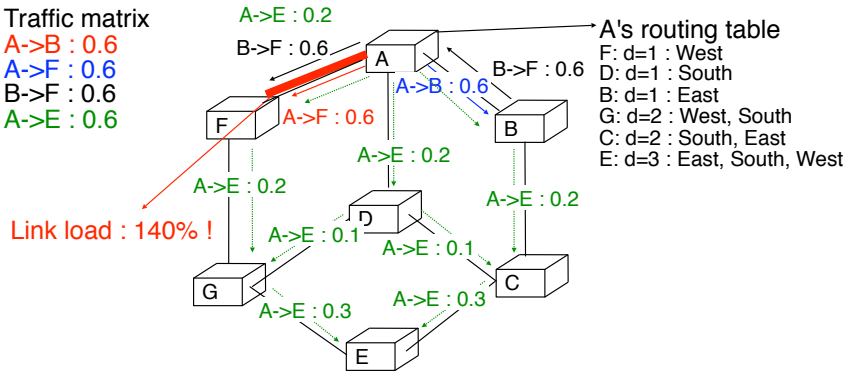
- Principle
  - Perform load balancing on a per TCP connection basis
  - Router identifies the connection to which each packet belongs and all packets from same connection are sent on same path
    - no reordering inside TCP connections
  - Issues to address
    - How to efficiently select the path for each TCP conn.
      - Router should not need to maintain a table containing
        - IP src, IP dest, Src port, Dest port : path to utilise
      - TCP connections towards to busy server should not be all sent sent on the same path

Per TCP connection load balancing (2)

- How to perform load balancing without maintaining state for each TCP connection ?
  - Principle
    - concatenate IP src, IP dest, IP protocol, Src port, and Dest port from the IP packet inside a bit string
      - bitstring = [IP src:IP dest:IP protocol:Src port:Dest port]
    - compute path = Hash(bitstring) mod P
      - hash function should be easy to implement and should produce very different numbers for close bitstring values
      - candidate hash functions are CRC, checksum, ...
  - Advantages
    - all packets from TCP connection sent on same path
    - traffic to a server will be divided over the links
  - Drawback
    - does not work well if a few TCP connections carry a large fraction of the total traffic
    - Polarisation issues if all routers use same hash

The polarisation problems happen if all routers use exactly the same hash as shown above. In this case, all routers of the network will compute the same hash and packets that are sent on the first interface by the first router will also be sent on the first interface by the second router... A possible way to avoid these polarisation issues

Limitations of Equal Cost Multipath



- Drawbacks of ECM
  - load balancing only works for *exactly equal* costs paths and few paths are exactly equal
  - local decision taken by each individual router

Extensions to Equal Cost Multipath have been proposed to allow routers to split traffic over non-shortest paths, but these extensions have not been implemented and deployed on real rotuers.

Some networks are designed to use a large number of equal cost paths to favour load balancing. Other networks are designed to avoid equal cost paths in order to ease management and debugging of problems.

IP-based network level traffic engineering

- How to improve the traffic distribution throughout the entire network ?
  - Principle
    - IGP link cost influences the utilisation of this link
      - Typical IGP link cost settings include
        - 1 for each link to select shortest path measured in hops
        - =link delay to select shortest path measured in seconds
        - f(bandwidth) to select shortest-high bandwidth path
        - example : $\frac{M}{link\ bandwidth}$
    - Careful selection of the IGP link costs to balance traffic
      - rerouting traffic outside a busy link by manually tweaking costs
      - optimising the flow of traffic instead a network for a given traffic matrix can considering it as a classical optimisation problem
        - Can be difficult if routers do not support ECM
        - possible with some restrictions when routers support ECM

CNPP/2008.10.

For example, the default OSPF link cost setting on Cisco routers is described in OSPF Design guide, available from http://www.cisco.com/warp/public/104/1.html

A method to optimally set the OSPF weights for a known traffic matrix was proposed in
B. Fortz and M. Thorup, Internet traffic engineering by optimizing OSPF weights, Proc. IEEE Infocom 2000, March 2000, available from http://www.ieee-infocom.org

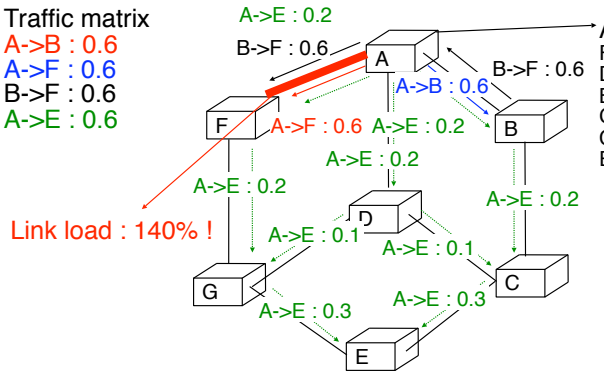
Other references include
Y.Wang, Z.Wang, and L.Zhang. Internet traffic engineering without full mesh overlaying. In *INFOCOM2001*, April 2001. available from http://www.ieee-infocom.org.
D.Lorenz, A.Orda, D.Raz, and Y.Shavitt. How good can IP routing be ? Technical Report 2001-17, DIMACS, May 2001. available from http://www.eng.tau.ac.il/~shavitt/pub/DIMACS01-17.ps.
Anja Feldmann, Albert Greenberg, Carsten Lund, Nick Reingold, and Jennifer Rexford. Netscope: Traffic engineering for ip networks. *IEEE Network Magazine*, March 2000.

Several tools have been developed to aid traffic engineering. Some commercial tools include :
Cariden http://www.cariden.com
WANDL http://www.wandl.com

as well as opensource tools :
http://totem.info.ucl.ac.be

IP-based network level traffic engineering (2)

Traffic matrix
A->B : 0.6
A->F : 0.6
B->F : 0.6
A->E : 0.6



A's routing table
F: d=1 : West
D: d=1 : South
B: d=1 : East
G: d=2 : West, South
C: d=2 : South, East
E: d=3 : East, South, West

- How to improve the traffic distribution ?
  - A should send traffic towards E only via its South port
  - B should send traffic towards F only via its South port
    - Possible by changing the IGP link costs

IP-based network level traffic engineering (3)

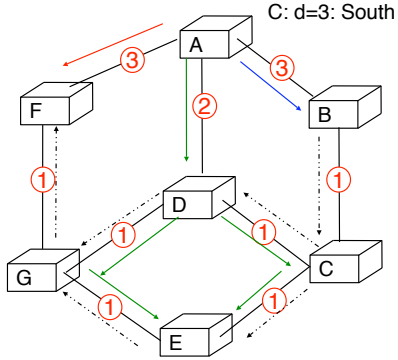
- Possible setting of the IGP link costs

D's routing table
A: d=2: North
B: d=2: South-East
C: d=1: South-East
E: d=2: S-E, S-W
F: d=2: S-W
G: d=1: S-W

A's routing table
F: d=3: West
D: d=2: South
B: d=3: East
E: d=4: South
G: d=3: South
C: d=3: South

B's routing table
A: d=3: North
C: d=1: South
D: d=2: South
E: d=E: South
F: d=4: South
G: d=3: South

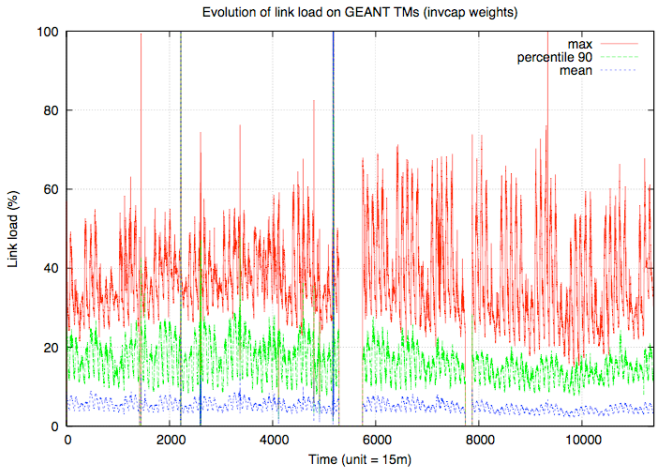
Traffic matrix
A->B : 0.6
A->F : 0.6
B->F : 0.6
A->E : 0.6



C's routing table
B: d=1: North
A: d=3: North-West
D: d=1: North-West
E: d=1: South-West
F: d=3: N-W, S-W
G: d=2:N-W, S-W

Changing IGP weights in GEANT

- Example : invcap



CNPP/2008.10.

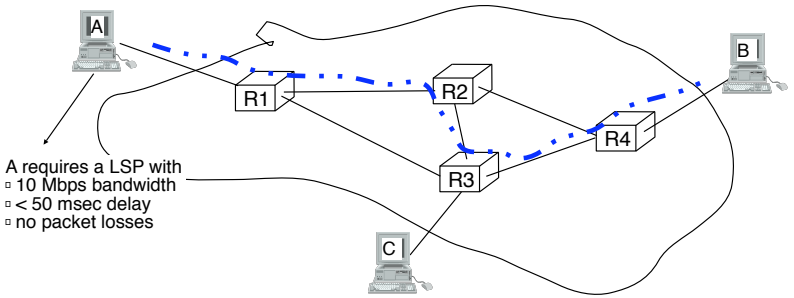
Other changes to the IGP weights are possible, of course.

MPLS-based traffic engineering

- Principle of the solution
  - Build a normal IP or IP+MPLS network
    - packet forwarding on shortest path towards destination
  - Collect traffic statistics at edge routers and information about link load
    - identify the most congested parts of the network
  - Ingress routers establish LSPs along a well chosen path to divert large traffic flows away from heavily loaded links
- Issues to solve
  - How can an ingress router determine an acceptable path for a given traffic flow ?
  - How to establish a LSP along a chosen path ?

Xipeng Xiao, Alan Hannan, Brook Bailey, and Lionel M. Ni. Traffic engineering with MPLS in the Internet. *IEEE Network Magazine*, March 2000.

Selecting a path with constraints



- How can we establish a LSP with QoS constraints through the network ?
  - Need information about capabilities of each link
  - Need an algorithm to select the best path according to specific constraints

Constrained routing

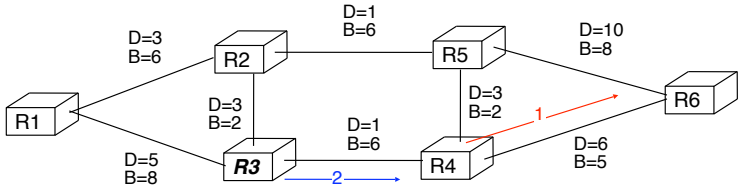
- What should be added to traditional routing algorithms ?
  - a way to distribute information about current network state
    - routers must know load of remote links to choose paths meeting constraints for flows with QoS guarantees
  - a way to compute a path subject to constraints
    - current routing algorithms find shortest path
    - how can we find a path with
      - minimum hop count
      - at least 10 Mbps
      - at most 10 msec of delay

Distributing load information

- Distance vector routing protocols [RIP,BGP]
  - routers conspire to distribute routing table
  - difficult to inform routers of load on remote links
  - difficult to support constrained routing
- Link state routing protocols [OSPF, IS-IS]
  - routers conspire to distribute network map
    - simple to add information about network load
  - routers distribute link state packets with load info
    - delay is already distributed as the IGP metric
    - bandwidth/link load is main information to distribute
    - tradeoff between frequent distribution (accurate information) and rare distribution (avoid network overload)
  - each router knows topology and load of each link and can find constrained paths

Distributing load information (2)

- Potential problem
  - Link load information is not distributed immediately
  - routers must establish flows based on partial information about the current load in the network



1. new flow [B=4] is created between R4 and R6
1. before information about load changes, R3 wants to create a new flow [B=2] towards R6
   - R3 believes that R3-R4-R6 is the best path

In this slide, B means bandwidth and D means Delay

Constraints

- Three types of constraints on path selection
  - Additive constraint
    - find path minimising
      - example
        - hop count
        - link delay or cost

$$\sum [d_1, d_2, ..., d_n]$$

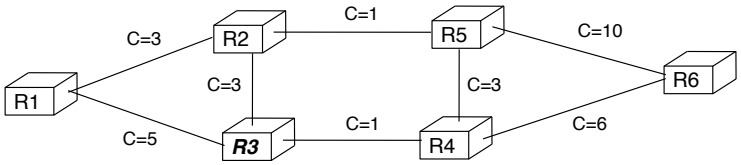
  - Multiplicative constraint
    - find path minimising
      - example
        - loss rate

$$\prod [d_1, d_2, ..., d_n]$$

  - Concave constraint
    - find path containing links whose characteristic is always above a given constraint
      - example
        - bandwidth
        - resource class or color

Finding a constrained path

- Single additive or multiplicative constraints
  - apply Dijkstra's algorithm
    - example

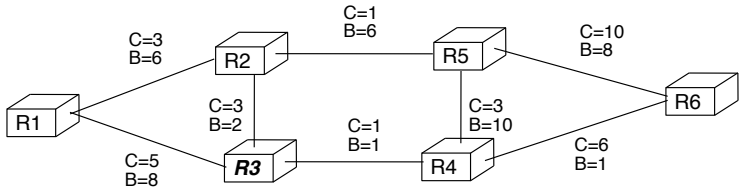


- 2 or more additive/multiplicative constraints
  - unfortunately problem is NP hard
    - need to evaluate all possible paths to find exact solution
    - several heuristics have been proposed in literature to find acceptable solutions

In the figure above, C is the IGP cost associated with each link. The Dijsktra algorithm builds a shortest path tree and is run on each router to determine the shortest path tree from this router to all routers inside the network. This tree is computed incrementally as follows.
First, the tree only contains the router which performs the computation
Then, all the routers that are adjacent to the router performing the computation are considered to be candidate and are put on the candidate list with costs equal to the cost of the links between this router and the candidate router. The candidate router with the smallest cost is added to the shortest path tree and removed from the candidate list.. All the neighbors of this router are then examined to see if a better path can be found for a candidate list. The candidate list is updated and the algorithm continues until all routers are added to the tree.
The shortest path tree from R3 is shown below :

Finding a constrained path (2)

- Concave constraints
  - fortunately easy to handle
    - remove from the network map all links that do not satisfy the constraint
    - use Dijkstra's algorithm on the reduced map
    - example
      - find shortest 3 Mbps from R3 to R6



CNPP/2008.10.

In the figure above, C is the IGP cost associated with each link and B is the available bandwidth on each link (in Mbps).

Another example would be to only utilise links offering supporting some kind of protection or to avoid satellite links.

Constrained routing in IP networks

- Several solutions proposed by researchers
  - Lessons learned
    - Constrained routing should be applied to flows and not on a per packet basis
      - Currently, constrained routing in IP networks is only used with MPLS to establish LSPs
    - Bandwidth and delay are key constraints
      - delay jitter is less important and difficult to efficiently support
    - Path selection should be performed by the source
      - the source of a flow selects an explicit route
      - intermediate nodes perform connection admission control but do not perform any constrained routing decision
        - path selection algorithm does not need to be standardised
        - if the new flow is acceptable, establishment continues otherwise the source will have to compute another path
  - Existing constrained routing protocols
    - OSPF-TE, ISIS-TE, PNNI (ATM)

CNPP/2008.10.

ISIS-TE is described in :

H.Smit and T.Li. IS-IS extensions for Traffic Engineering (RFC 3784) updated by RFC 4205
Intermediate System to Intermediate System (IS-IS) Extensions in Support of Multi-Protocol Label Switching (GMPLS) (RFC 4205)

OSPF-TE is defined in

D. Katz, K. Kompella, D. Yeung, Traffic Engineering (TE) Extensions to OSPF Version 2 (RFC 3630)
PNNI is described in

ATM Forum. Private Network-Network Interface specification version 1.0 (PNNI 1.0). ATM Forum specification af-pnni-0055.000, March 1996.

Additional information about QoS routing protocols may be found in :
Shigang Chen and Klara Nahrstedt. An overview of quality of service routing for next-generation high-speed networks: Problems and solutions. *IEEE Network Magazine*, 12(6):64--79, November 1998.

G.Apostolopoulos, R.Guerin, S.Kamat, A.Orda, A.Przygienda, and D.Williams. QoS routing mechanisms and OSPF extensions. Internet Draft, draft-guerin-qos-routing-ospf-04.txt, Internet Engineering Task Force, December 1998. Work in progress.

E.Crawley, R.Nair, B.Rajagopalan, and H.Sandick. RFC 2386: A framework for QoS-based routing in the Internet, August 1998. Status: INFORMATIONAL

OSPF-TE
An example constrained routing protocol

- Extension to OSPF designed to aid in the establishment of traffic engineered LSPs
  - OSPF-TE distributes new info about each link
    - link type and link Id
    - local and remote IP addresses
    - traffic engineering metric
      - additional metric to specify the cost of this link
    - maximum bandwidth
      - maximum amount of bandwidth usable on this link
    - maximum reservable bandwidth
      - maximum amount of bandwidth that can be reserved by LSPs
    - unreserved bandwidth
      - amount of bandwidth that is not yet reserved by LSPs
    - resource class/color
      - can be used to specify the type of link (e.g. Expensive link would be colored in red and cheap links in green)

OSPF-TE is described in :

D.Katz and D.Yeung. Traffic Engineering extensions to OSPF.RFC3640

Using RSVP to distribute MPLS labels

- Principle
  - RSVP supports downstream on-demand label allocation
    - RSVP extension for MPLS called RSVP-TE
  - Ingress LSR sends PATH message towards egress LSR
    - PATH message includes Label Request Object
  - Egress LSR sends label back in RESV message
    - RESV propagates the labels hop-by-hop

RSVP-TE is defined in the following documents :

RFC3209 RSVP-TE: Extensions to RSVP for LSP Tunnels. D. Awduche, L. Berger, D. Gan, T. Li, V. Srinivasan, G. Swallow. December 2001. (Format: TXT=132264 bytes) (Status: PROPOSED STANDARD)

RFC3210 Applicability Statement for Extensions to RSVP for LSP-Tunnels. D. Awduche, A. Hannan, X. Xiao. December 2001. (Format: TXT=17691 bytes) (Status: INFORMATIONAL)

Besides RSVP, a second protocol which is an extension to LDP can be used to establish LSPs for traffic engineering purposes :

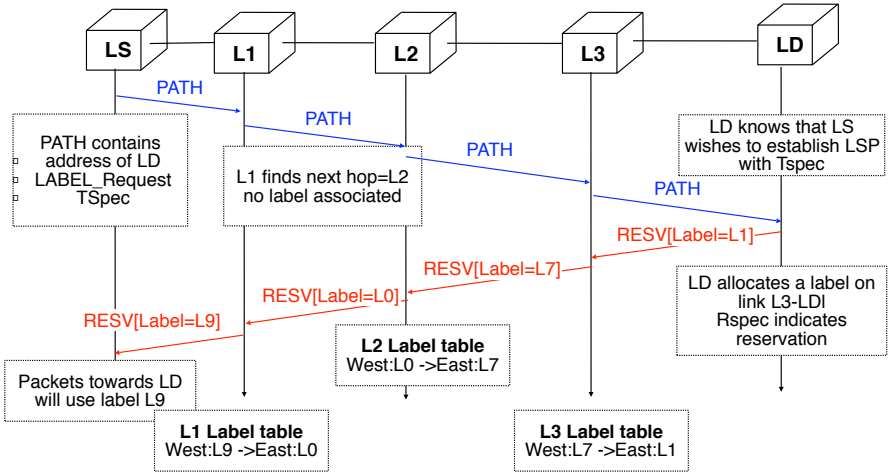
RFC3212 Constraint-Based LSP Setup using LDP. B. Jamoussi, Ed., L. Andersson, R. Callon, R. Dantu,. January 2002. (Format: TXT=87591 bytes) (Status: PROPOSED STANDARD)

RFC3213 Applicability Statement for CR-LDP. J. Ash, M. Girish, E. Gray, B. Jamoussi, G. Wright. January 2002. (Format: TXT=14489 bytes) (Status: INFORMATIONAL)

RFC3214 LSP Modification Using CR-LDP. J. Ash, Y. Lee, P. Ashwood-Smith, B. Jamoussi, D. Fedyk, D. Skalecki, L. Li. January 2002. (Format: TXT=25453 bytes) (Status: PROPOSED STANDARD)

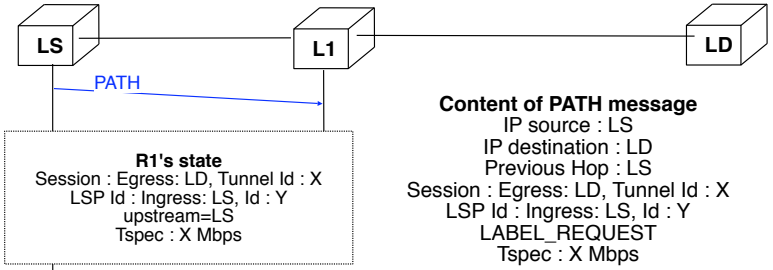
Using RSVP to distribute MPLS labels (2)

- LSP establishment with RSVP-TE



RSVP-TE : detailed example

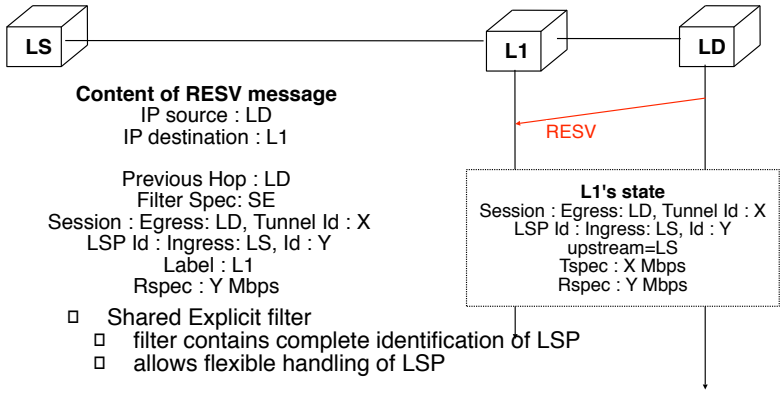
- What happens inside the LSR ?



- Tunnel Id
  - identifies the LSP being established between LS and LD
- LSP Id
  - identifies an "instance" of the LSP being established
- Label Request
  - indification of the network protocol (IP, Ipv6) carried on LSP

RSVP-TE : detailed example (2)

- Content of the RESV message



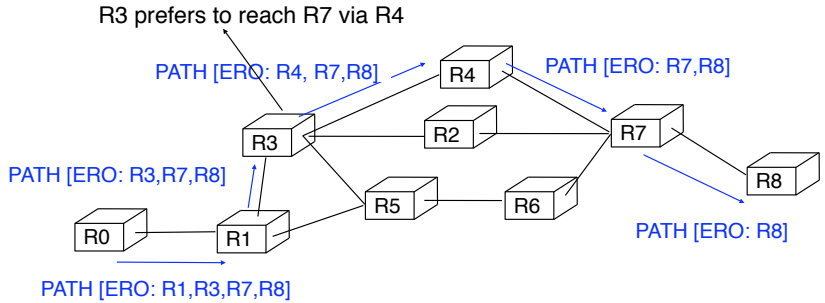
- Shared Explicit filter
  - filter contains complete identification of LSP
  - allows flexible handling of LSP

Identification of LSP : [Ingress, Egress, Tunnel ID, LSP Id]

RSVP-TE : Explicit Routes

- How to establish LSP along non-shortest path ?
  - Solution
    - Ingress LSR may specify the route to be followed by an LSP being established
    - Route specification is composed of a list of
      - IP addresses
      - Subnet prefixes
      - Autonomous System numbers
    - Two types of route specifications
      - Strict route
        - the LSP must pass through each LSR specified by ingress LSR
      - Loose route
        - the LSP can pass through non-specified LSR between two specified LSRs

RSVP-TE : Explicit Routes (2)



CNPP/2008.10.

When RSVP-TE is used to establish LSPs, this is usually inside a single AS. In this case, LSPs are established between loopback addresses of routers and usually a full-mesh of LSPs is used. Some providers also use several LSPs between pairs of important routers to

- perform load balancing,
- ensure that one LSP will always remain available even in case of link failures
- support different types of service (e.g. voice, normal Internet) over different paths

There are some deployments with RSVP-TE across ASes. In this case, the ERO can specify a prefix or AS instead of specifying an IP address.

MPLS for traffic engineering

- Principle
  - ingress LSR establishes LSPs (with guaranteed resources) along chosen paths
- Issues to consider
  - How to reserve bandwidth for a given LSP ?
    - Rely on Tspec and Rspec
  - How to support varying traffic flows ?
    - It should be possible to dynamically modify the LSP resources. If there are not enough resources to support an increase, the LSP should keep the old resources
  - How to dynamically reroute an established LSP ?
    - For example more bandwidth is available on another path or because one link used by the LSP failed

RSVP-TE : Resource increase

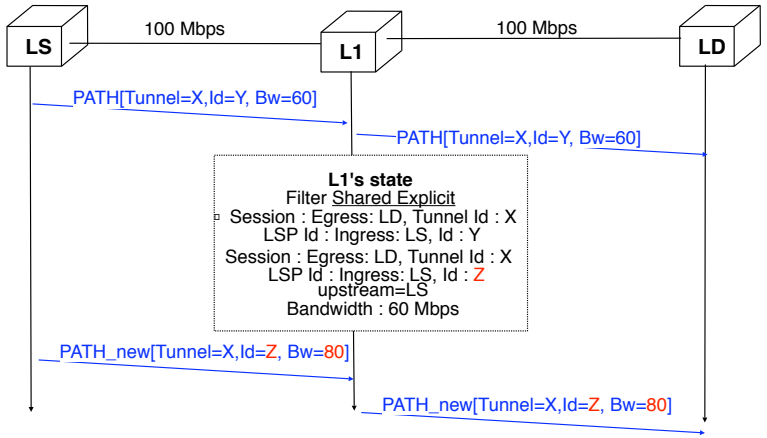
- How to smoothly increase the bandwidth of an LSP ?
  - Simple solutions

1. Change resources in PATH and RESV messages
   - if there are not enough resources available in the network to support the bandwidth increase, network will send RESVErr and entire LSP will be removed from network
   - not suitable for important LSP

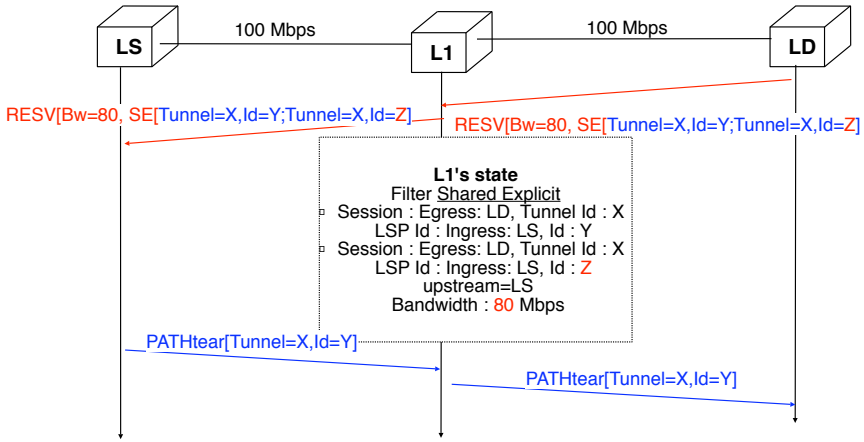
2. Try to establish new LSP
   - create new LSP and once accepted remove old LSP
   - drawback : the new LSP might be rejected due to the resources already used by the existing LSP

RSVP-TE : Resource increase (2)

- Smooth resource increase



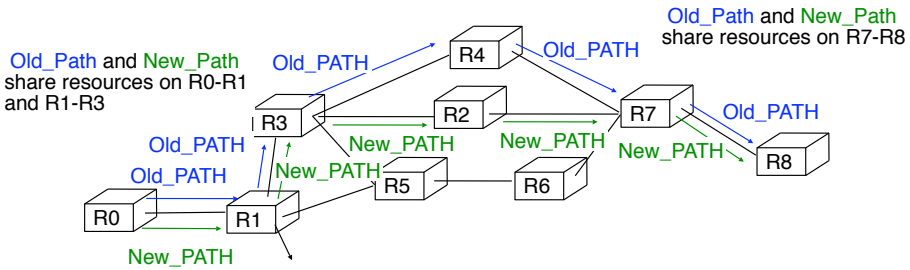
RSVP-TE : Resource increase (3)



CNPP/2008.10.

RSVP-TE : Changing routes

- How to change the route of an explicitly routed LSP ?
  - Same principle as for resource increase

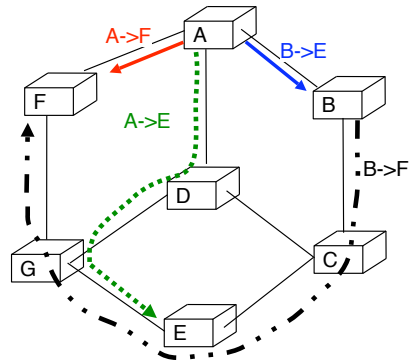


Differences between Old_Path and New_Path
- ERO object
- LSP Id

Traffic engineering with MPLS

A's routing table
F: d=1 : West
D: d=1 : South
B: d=1 : East
G: d=2 : West
C: d=2 : South
E: d=3 : South

A's mapping table
Destination Label
B LB
E LE
F LF



B's mapping table
Destination Label
F LF

Traffic matrix
A->B : 0.6
A->F : 0.6
B->F : 0.6
A->E : 0.6

B's routing table
A: d=1 : North
C: d=1 : South
D: d=2 : South
E: d=2 : South
F: d=2 : North
G: d=3 : South

MPLS
MultiProtocol Label Switching

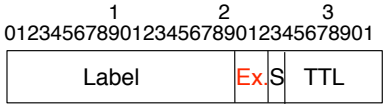
- Plan
  - Multiprotocol Label Switching
    - The label swapping forwarding paradigm
    - Integrating label swapping and IP
  - Utilisation of MPLS
    - Destination based packet forwarding
    - Simpler ISP backbones
    - Traffic engineering
    - QoS support
    - Fast restoration

MPLS and IP QoS

- What could be the benefit of MPLS to support IP QoS ?
  - With differentiated or integrated services, the path followed by IP packets is independent of their QoS since those architectures did not change routing
  - When MPLS is used, IP packets with distinct QoS requirements may be placed inside distinct LSPs that follow different paths inside the network
    - MPLS allows to utilize distinct routes for packets with distinct QoS requirements

MPLS and Differentiated Services

- How can we support Differentiated services inside a MPLS network ?
- LSR must know QoS required by each packet
  - Two complementary methods
    - Indicate the QoS required by all packets of a given LSP at LSP establishment time (e.g. With special RSVP objects)
    - Encode QoS info inside MPLS label



  - EXP field is part of MPLS header and contains 3 bits
  - special RSVP objects can be used to map Exp field with DSCP

CNPP/2008.10.

The MPLS support of Diffserv is discussed in :

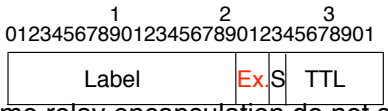
F.Le Faucheur, L.Wu, B.DavieS. Davari, P.Vaananen, R.Krishnan, P.Cheval, and J.Heinanen. Mpls support for differentiated services. Internet draft, draft-ietf-mpls-diff-ext-09.txt, work in progress, April 2001.

Other information may be found in :

S.Ganti, S.Seddigh, and B.Nandy. Mpls support of differentiated services using e-lsp. Internet draft, draft-ganti-mpls-diffserv-elsp-00.txt, work in progress, April 2001.

Diffserv support with MPLS

- The "*IP*" way, aka E-LSPs
  - a single LSP may carry packets receiving several differentiated services
  - each MPLS router relies on EXP header field to determine the service for each received packet
    - EXP field is part of MPLS header and contains 3 bits



    - ATM/frame relay encapsulation do not support 3 bits
  - useful to reduce the number of required LSPs in large networks

Diffserv support with MPLS (2)

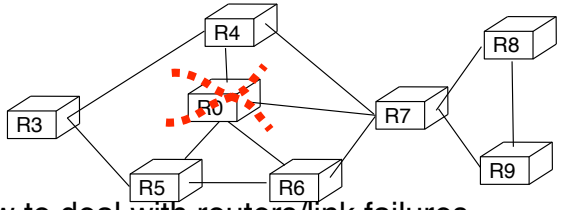
- The "*ATM*" way, aka L-LSPs
  - one LSP carries packets receiving a single service
    - the EXP field of the MPLS header may be used to specify the drop preference for each packet (e.g. For AF)
    - MPLS router decides service for a receiving packet based on lable value
  - the service used by the LSP is specified at LSP establishment time
  - useful in ATM/frame relay networks or when number of LSPs is not a constraint
  - each L-LSP may have its own explicit route
  - more L-LSPs than E-LSPs will be needed

MPLS
MultiProtocol Label Switching

- Plan
  - Multiprotocol Label Switching
    - The label swapping forwarding paradigm
    - Integrating label swapping and IP
  - Utilizations of MPLS
    - Destination based packet forwarding
    - Simpler ISP backbones
    - Traffic engineering
    - QoS support
    - Fast restoration

Dealing with link/router failures

- Providing a good service means
  - providing the promised bandwidth/delay ...
    ... even in case of network failures



- How to deal with routers/link failures
  - Detect the failed component
    - Detection by the routing protocol itself (adjacency information)
    - Detection with the help of layer 2 information (carrier lost)
  - Propagate the bad news inside the network

Detecting link/router failures

- How to detect link/router failures ?
  - Rely on failure detection of physical layer
    - possible with SONET/SDH and some types of modems
    - detection delay may vary from 10 msec to a few seconds
    - difficult on LAN interfaces such as Ethernet
  - Relying on the routing protocols
    - BGP
      - once peering session has been established, KEEPALIVE messages are sent every 30 seconds
      - peer is dead if no KEEPALIVE within 90 seconds
    - OSPF
      - Hello packets sent every 10 seconds
      - Neighbor is dead if no hello received in 40 seconds
    - RIP
      - router sends routing table every 30 seconds
      - neighbor is dead if no table within the 180 seconds

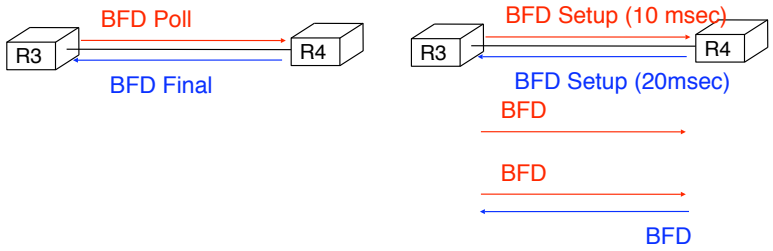
Untuned routing protocols may need 10s seconds to detect failures !

Newer implementations of RIPv2 should support triggered updates. With these triggered updates, a bad news (i.e. The metric of a route being set to infinity to indicate a link failure) may be sent immediately. However, once a triggered update has been received, a router should not send another update for the same route within a period of 1 to 5 seconds. This dampening process is used to reduce the network load, but it may also increase the convergence time inside the network.

Most implementations allow to configure the keepalive timers used in each routing protocol. However, care must be taken when setting a keepalive timer to a too low value. For example, consider BGP. If in theory a keepalive timer of 1 millisecond would be interesting to quickly detect failures, in practice, using such a timer would create problems. Since BGP is implemented as an application running on the central CPU of the router which is also supporting SNMP, RIP/OSPF, console acces, ... it is usually impossible to guarantee that BGP would be able to send a KEEPALIVE every 1 milliseconds. Furthermore, even if BGP had a dedicated CPU, BGP relies on TCP and the loss of one TCP segment caused by random transmission errors could force TCP to wait until the retransmission of the lost segments to deliver the KEEPALIVE message.

Detecting link/router failures

- Another solution
  - Bidirectionnal forwarding detection (BFD)
  - Simple UDP based protocol
    - can be implemented on router linecards directly
  - Two modes of operation



- O(10 msec) failure detection becomes possible

CNPP/2008.10.

Bidirectionnal Forwarding Detection (BFD) is intended to be a generic protocol able to detect that a link is up and running with a much shorter detection time that the routing protocols. BFD supports several modes of operations , see :

D. Katz, D. Ward, Bidirectional Forwarding Detection, Internet draft, work in progress, Feb 2009 http://www.ietf.org/internet-drafts/draft-ietf-bfd-base-09.txt

Dealing with link/router failures

- How long does it take to propagate the failure information inside the network ?
  - O(a few sec.) inside a single AS
    - some features of routing protocols implementations may introduce additional delays
      - a popular router OS introduces a default delay of five seconds between the reception of an OSPF routing update and the computation of the updated routing table
        - this reduces CPU utilization but increases convergence time
    - a convergence time of around half a second is possible today in large networks
  - O(tens sec.) across the Internet
    - measurements on the Internet show that a failure is recovered within several tens of seconds, but in some cases it may take several minutes

Several groups are working on improving the convergence of IGP protocols. These improvements include the development of algorithms to incrementally update the IGP routing table and tuning of the timers used by a given implement.
For more information, see e.g. :

C.Alaettinoglu, V.Jacobson, and H.Yu. Towards millisecond IGP congergence. Internet draft, draft-alaettinoglu-ISIS-convergence-00.ps, work in progress, November 2000.

Cengiz Alaettinoglu and Steve Casner, ISIS Routing on the Qwest Backbone: a Recipe for Subsecond ISIS Convergence, NANOG 24, http://www.nanog.org/mtg-0202/cengiz.html, February 2002

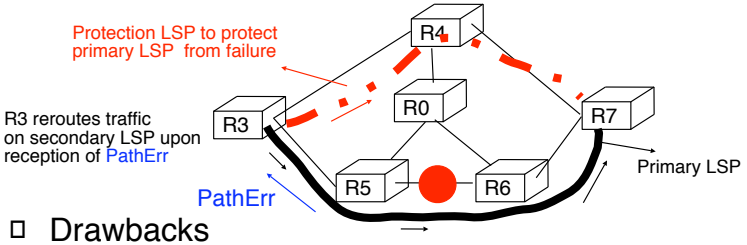
A. Retana, IP Routing protocol scalability - theory and examples, 2001, NANOG http://www.nanog.org/mtg-0110/retana.html

MPLS based fault-tolerance

- Can we do better with MPLS ?
  - MPLS forwarding does not depend on routing
    - a MPLS router can forward packets even when routing has not converged provided that a secondary LSP exist
  - Solution
    - Establish secondary LSPs to protect important LSPs
      - secondary LSP is established and maintained inside the network but carries traffic only in case of failure of the primary LSP
    - when an outgoing link or router fails inform headend LSRto stop using the primary LSP and switch all traffic to the protection LSP
      - this operation can be done by the MPLS router itself without any cooperation with other routers provided the protection LSP exists

End-to-End secondary LSP

- First solution
  - Secondary LSP established between ingress LSR and egress LSR
  - In case of failure, PathErr message sent to ingress



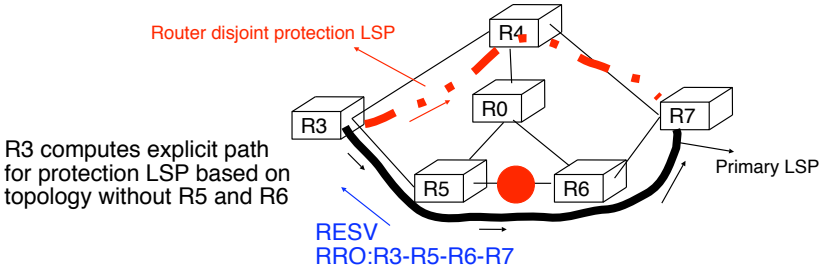
  - Drawbacks
    - PathErr may take some time to reach ingress LSR
      - packets may be lost between failure detection and reception of PathErr
    - one secondary LSP for each primary LSP

Selection of path for secondary LSP

- How to select a suitable path for a secondary LSP to protect a primary LSP ?
  - Principle of the solution
    - Ingress LSR selects path for primary LSP by using its path selection algorithm on the entire network topology
    - Knowing the path of primary LSP, ingress LSR computes a path for the secondary LSP by using its path selection algorithm on the network topology where the resources used by primary LSP have been removed
      - secondary LSP must rely on different physical resources than primary LSP
      - this implies that the ingress LSR needs to know the physical resources used by each LSP
        - RSVP carries this information in the RRO object

Selection of path for secondary LSP (2)

- Types of end-to-end protection LSPs
  - router-disjoint protection LSP
    - does not utilize the same transit routers as the primary
  - link-disjoint protection LSP
    - does not utilize the same links as the primary LSP



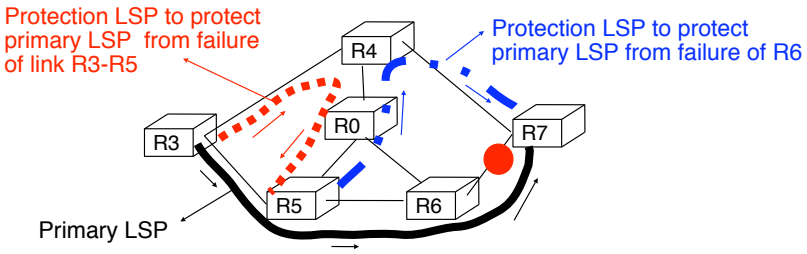
R3 computes explicit path for protection LSP based on topology without R5 and R6

Improving MPLS protection

- Issues with end-to-end protection LSPs
  - one protection LSP must be established for each primary LSP
    - doubles the number of LSPs in the network
    - if bandwidth is reserved for primary, bandwidth must be reserved for secondary
  - failure information must travel back to ingress LSR
- It would be interesting to
  - Reduce the time required to switch to the protection LSP
  - Reduce the number of protection LSPs established inside the network

Failures to consider

- Two types of failures
  - Router failure
    - a router and all the links attached to this router fail
  - Link failure
    - a link fails between two adjacent routers



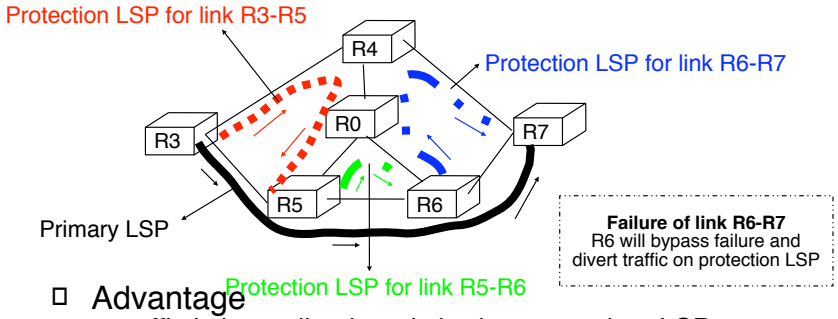
CNPP/2008.10.

In pratice, it might also be useful to group LSPs from groups of failures. For example, consider the utilization of different wavelengths over the same optical fiber. In the slide above, links R5-R0 and R5-R6 could travel over the same physical fiber. If this fiber fails, then both R5-R0 and R5-R6 would fail together. In this case, if the primary LSP uses link R5-R6, then the secondary LSP should not also utilize link R5-R0.

It is possible to deal with those issues by advertising in the IGP the physical ressources that are used by each link, but these details are outside the scope of this presentation.

Protecting a complete path

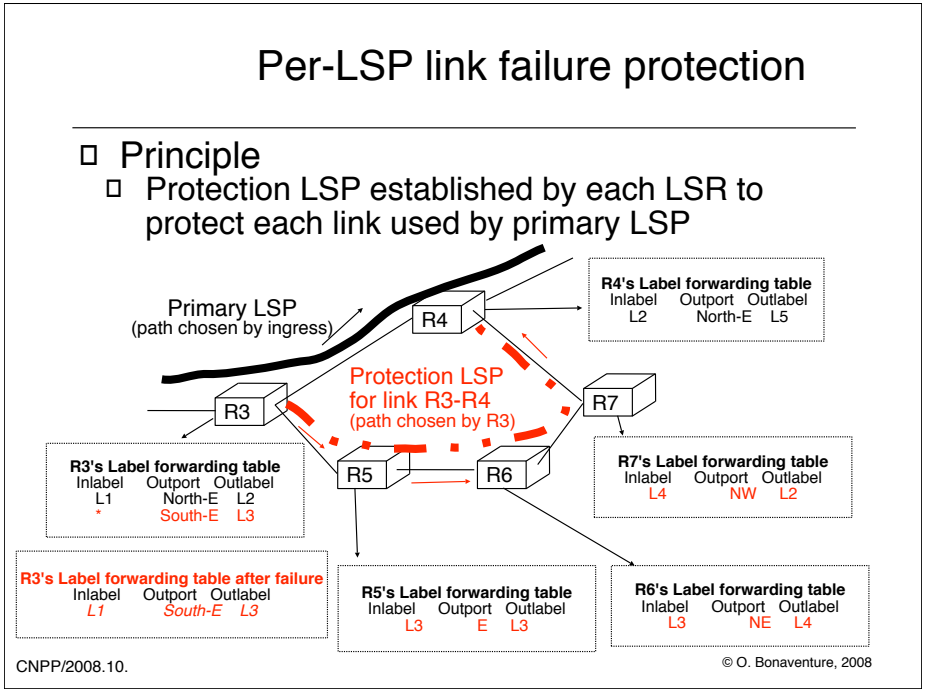
- Principle
  - one protection LSP is used to protect each link



- Advantage
  - traffic is immediately switched to secondary LSP
- Drawback
  - a large number of protection LSPs may be required

In this slide, we assume that link protection was requested.Those protection LSPs are called detour LSPs in the standardization documents.

P. Pan, D. Gan, G. Swallow, J. Vasseur, D. Cooper, A. Atlas, M. Jork, Fast Reroute Extensions to RSVP-TE for LSP Tunnels, Internet draft, draft-ietf-mpls-rsvp-lsp-fastreroute-00.txt, work in progress, Jan 2002



Extensions to RSVP-TE have been proposed to allow the ingress LSR to request the establishment of automatic link protection LSPs by each LSR on the path of a primary LSP. See

P. Pan, D. Gan, G. Swallow, J. Vasseur, D. Cooper, A. Atlas, M. Jork, Fast Reroute Extensions to RSVP-TE for LSP Tunnels, Internet draft, draft-ietf-mpls-rsvp-lsp-fastreroute-00.txt, work in progress, Jan 2002

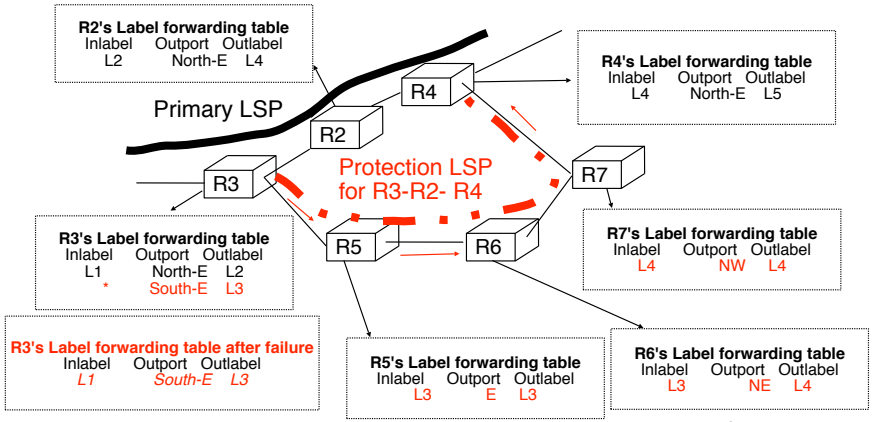
The ingress LSR, when establishing its LSP may request the LSP to be protected against link failures or router failures.

R3 computes the path for the protection LSP by finding in the network the best path to reach R4 without using link R3-R4.

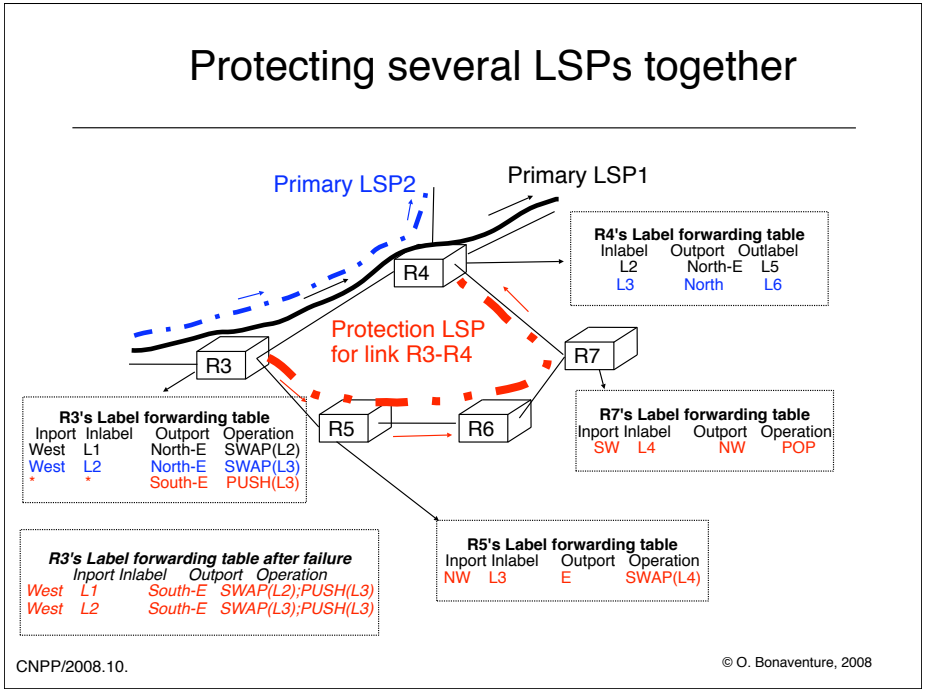
Note that in this example, we use per-LSR label space. This allows R4 to send the packets received with label L2 on the primary LSP, independently of the interface from which the packets were received (from R3 on the primary or from R7 on the detour LSP). For this, R4 simply suggests the utilization of label L2 in the RESV message that it sends upon reception of the Path message sent by R3 to establish the detour LSP to protect the primary LSP from the failure of link R3-R4.

Per-LSP router failure protection

- Principle
  - Protection LSP established by each LSR



CNPP/2008.10.



This type of protection LSP is also called a bypass tunnel

Extensions to RSVP-TE have been proposed to allow the ingress LSR to request the utilization of bypass tunnels by each LSR on the path of a primary LSP to be protected. See

P. Pan, D. Gan, G. Swallow, J. Vasseur, D. Cooper, A. Atlas, M. Jork, Fast Reroute Extensions to RSVP-TE for LSP Tunnels, Internet draft, draft-ietf-mpls-rsvp-lsp-fastreroute-02.txt, work in progress, Feb 2003

In this example, R3 establishes a bypass tunnel to protect LSP1 and LSP2 from the failure of link R3-R4. For this, R3 creates a protection LSP to reach R4 without using link R3-R4. If link R3-R4 fails, then R3 will update its label forwarding table to first put the same labels as on the primary LSP and then push label L3 on top of those labels. Packets from LSP1 and LSP2 will then be sent to router R7 that will POP the top label before forwarding the packets towards R4.

We assume in the example that R4 uses a per-LSR label space. Note also that R7 is removing (POP) the label of the bypass tunnel before sending packets to R4. This allows R4 to receive the packets with the same labels as on the primary LSP.