

# Modelos de Interconexão IP-ATM

Eleri Cardozo  
Mauricio F. Magalhães

2 de setembro de 2004



# Sumário

<b>1</b>	<b>Modelos ATM para Interconexão de Redes</b>	<b>5</b>
1.1	Introdução . . . . .	5
1.1.1	Terminologia . . . . .	6
1.2	TCP/IP Sobre ATM . . . . .	7
1.3	Modelos ATM para Interconexão de Redes . . . . .	9
1.3.1	O Modelo Overlay . . . . .	13
1.3.2	O Modelo Peer . . . . .	14
1.4	Considerações Sobre Roteamento . . . . .	15
1.4.1	O roteamento IP e o Modelo <i>Overlay</i> . . . . .	16
1.4.2	O Roteamento IP e o Modelo <i>Peer</i> . . . . .	19
1.5	Chaveamento de Pacotes Orientado a Fluxo . . . . .	21
1.6	Chaveamento de Pacotes Orientado a Topologia . . . . .	24
<b>2</b>	<b>MPLS - Multiprotocol Label Switching</b>	<b>29</b>
2.1	Características Gerais . . . . .	30
2.2	Arquitetura MPLS . . . . .	34
2.3	Conceitos Básicos . . . . .	36
2.3.1	Mecanismos de Alocação de Rótulos . . . . .	51
2.3.2	Detecção de <i>loops</i> . . . . .	57
2.3.3	Seleção de Rota . . . . .	58



# Capítulo 1

## Modelos ATM para Interconexão de Redes

### 1.1 Introdução

Uma infra-estrutura de comutadores ATM, em geral denominada de nuvem ATM, deve ser entendida como um ambiente sobre o qual vários serviços serão oferecidos, serviços estes correspondendo ao tráfego de dados, vídeo e voz associados às aplicações dos usuários. Como forma de viabilizar a oferta dos serviços através desta rede ATM, a qual se espera formada por comutadores de fabricantes diversos, há a necessidade da padronização dos modelos de serviços para implementação nos equipamentos dos fabricantes. Esta demanda motivou os esforços de instituições e grupos como o ITU-T (*International Telecommunication Union*), ATM Forum e IETF (*Internet Engineering Task Force*) no desenvolvimento de padrões voltados para a utilização da tecnologia ATM como infra-estrutura de interconexão de redes.

Os serviços a serem oferecidos em uma nuvem ATM podem ser classificados como serviços voltados para a interconexão de redes de computadores para fins de troca de dados e serviços com características de fluxo contínuo, como no caso da transmissão de voz e de vídeo. Estes serviços possuem características de tráfego distintas e, no entanto, devem ser suportados por uma mesma infra-estrutura ATM. O interesse principal deste relatório consiste no estudo dos modelos que estão sendo propostos para a interconexão de redes de computadores. Esta classe de serviços pode ainda ser subdividida nos modelos voltados para a interconexão de redes locais e aqueles voltados para as redes de longa distância. No primeiro caso, temos como exemplos a emulação de LANs sobre ATM, o IP Clássico sobre ATM, *Multi-*

## 6CAPÍTULO 1. MODELOS ATM PARA INTERCONEXÃO DE REDES

*protocol Over ATM* (MPOA) e *Multiprotocol Label Switching* (MPLS). Estes 2 últimos modelos também podem ser vistos como soluções que possuem escalabilidade para serem utilizados no âmbito de redes de longa distância. No segundo caso, os modelos propostos para o suporte dos serviços Frame-Relay e SMDS/CBDS sobre ATM são os mais importantes.

Como é comum nos processos de padronização, muitas vezes o mercado surge com soluções de fabricantes que tentam se impor como padrões *de facto* em reação às soluções, muitas vezes complexas, propostas pelos grupos de padronização. No caso da interconexão de redes de computadores através de ATM podemos destacar soluções do tipo IP-Switching (Ipsilon), Tag-Switching (Cisco), IP-Navigator (Ascend) e Aris (IBM), dentre outras, que surgem como alternativas às soluções propostas pelo IETF e ATM Forum.

### 1.1.1 Terminologia

Na discussão a seguir é importante estabelecer a terminologia a ser utilizada de modo a contribuir para uma melhor compreensão do texto. No caso de uma nuvem ATM, um sistema ou dispositivo intermediário deve ser entendido como um comutador ATM, enquanto um sistema final (*end-system*) pode ser uma chave LAN (*LAN switch*), um processador diretamente conectado à nuvem ATM (PC, estação de trabalho, etc) denominado *estação ATM*, ou ainda um roteador. Este último é normalmente considerado nas redes tradicionais como um elemento intermediário no nível da camada de rede. Sistemas finais especialmente importantes são os dispositivos de interconexão de redes, via de regra não-ATM (roteadores e chaves LAN). Estes dispositivos são denominados *dispositivos de borda* e, tomando-se como base um pacote de rede que irá trafegar pela nuvem, o dispositivo de borda pode ser um dispositivo de ingresso (entrada do pacote na nuvem) ou de egresso (saída do pacote da nuvem). A figura 1.1 ilustra estes termos.

Do ponto de vista do *multicasting*, ou seja, um mesmo pacote é encaminhado a múltiplos destinos, uma subrede pode naturalmente suportar esta facilidade como no caso das redes onde há compartilhamento de um meio comum tal qual nas redes locais. Caso a subrede seja incapaz de encaminhar pacotes em difusão (*broadcast*) sem a utilização de servidores adicionais, a subrede é denominada de NBMA (*Non-Broadcast Multiple Access*). Como exemplos de redes NBMA temos as redes X.25, Frame-Relay e ATM.

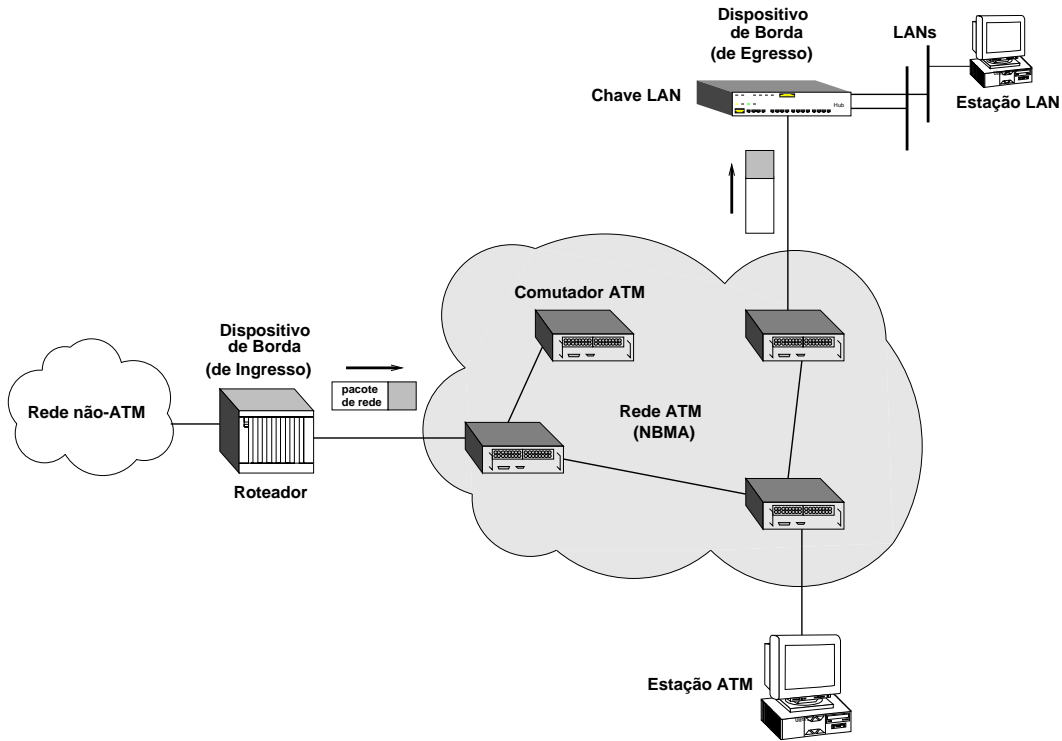


Figura 1.1: Terminologia para interconexão de redes via ATM

## 1.2 TCP/IP Sobre ATM

Para melhor entendermos os conceitos referentes ao transporte de datagramas IP sobre uma rede ATM, um exame mais detalhado da camada interface de rede da arquitetura TCP/IP se faz necessário. A camada interface de rede na arquitetura TCP/IP (figura ??) agrega uma infra-estrutura necessária ao transporte de datagramas. Esta infra-estrutura deve contemplar *pele menos* as duas primeiras camadas do modelo OSI sendo que, em alguns casos, ela corresponderá às três camadas inferiores do modelo OSI.

Em ambientes de redes locais, a camada interface de rede se constitui exatamente das camadas físicas e de enlace (incluindo a subcamada de acesso ao meio). Assim sendo, em ambientes locais a primeira camada da arquitetura TCP/IP oferece a infra-estrutura para o transporte de datagramas entre duas estações da rede local.

Em ambientes de longa distância, temos duas variantes. A primeira va-

## 8CAPÍTULO 1. MODELOS ATM PARA INTERCONEXÃO DE REDES

riante se refere a conexões ponto-a-ponto como, por exemplo, uma conexão E1 com o protocolo de enlace PPP (*Point-to-Point Protocol*). Nesta configuração, temos uma infra-estrutura que agrega as duas primeiras camadas do modelo OSI sendo, portanto, idêntica àquela de ambientes locais. A segunda variante diz respeito às redes de longa distância que incorporam uma camada 3 específica. Estas redes apresentam serviços de camada 3 tais como roteamento, endereçamento e encaminhamento de pacotes, serviços estes também oferecidos pela camada inter-redes (IP) da arquitetura TCP/IP. A diferença fundamental entre estas duas variantes é que, no primeiro caso, o roteamento ocorre somente na camada inter-rede da arquitetura TCP/IP enquanto, no segundo caso, existe uma sobreposição de funções conforme ilustrado na figura 1.2, onde a rede de longa distância é representada por uma nuvem ATM. Quando a rede de transporte possui serviços de camada 3, a sobreposição de funções é inevitável e determinante nas soluções de interconexão entre a rede TCP/IP e a rede de transporte. Na sequência, esta sobreposição de funções será discutida do ponto de vista do encaminhamento (*forwarding*) e do roteamento/endereçamento de datagramas.

O encaminhamento de datagramas IP em redes que possuem serviço de camada 3 tais como X.25 e ATM é uma tarefa aparentemente simples: datagramas IP são transportados no campo de dados dos pacotes da rede de transporte. Seja o exemplo da figura 1.3 onde uma conexão estabelecida através da rede ATM interliga dois roteadores. No exemplo, datagramas IP são *encapsulados* em unidades de protocolo (PDU) da camada de adaptação tipo 5 (AAL5). Entretanto, um problema que se coloca é como o roteador que recebe uma PDU da AAL5 “sabe” que a PDU transporta um datagrama IP ? Em outras palavras, como multiplexar o tráfego de diferentes protocolos numa única conexão de rede ? A solução comumente empregada é adicionar um identificador imediatamente antes do início do datagrama. Este identificador, denominado SNAP (*Subnetwork Access Protocol*) é detalhado na seção ??.

Quanto a sobreposição das funções de roteamento e endereçamento, duas possibilidades podem ser exploradas. A primeira concentra-se na coexistência entre as duas formas de roteamento e endereçamento: uma forma para a rede IP e outra para a rede de transporte. Seja ainda a figura 1.3, quando o roteador R1 decide encaminhar um pacote ao roteador R2 (roteamento IP), o mesmo encapsula o pacote numa PDU da AAL5 e a envia através da conexão ATM. No interior da rede ATM, a rota da conexão foi determinada por ocasião de seu estabelecimento via outra estratégia de roteamento (PNNI), estratégia esta totalmente desvinculada do roteamento IP.



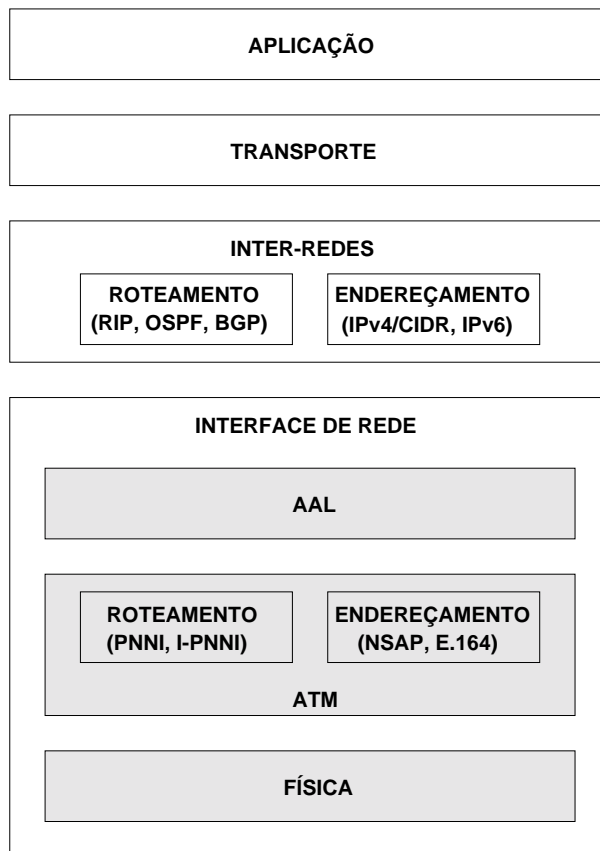


Figura 1.2: Sobreposição de funções TCP/IP e ATM

A segunda possibilidade baseia-se unificação do roteamento e endereçamento. Via de regra, esta estratégia mantém o roteamento e endereçamento IP, eliminando o roteamento e endereçamento da rede de transporte. Estas duas possibilidades são detalhadas a seguir no âmbito dos modelos *Overlay* e *Peer*.

### 1.3 Modelos ATM para Interconexão de Redes

A utilização inicial reservada para a tecnologia ATM foi a de *backbone* de alta velocidade para comportar o tráfego de dados entre redes locais e no transporte de dados em redes de longa distância. Entretanto, ATM é utilizado na atualidade também em outros cenários, desde redes locais de alta

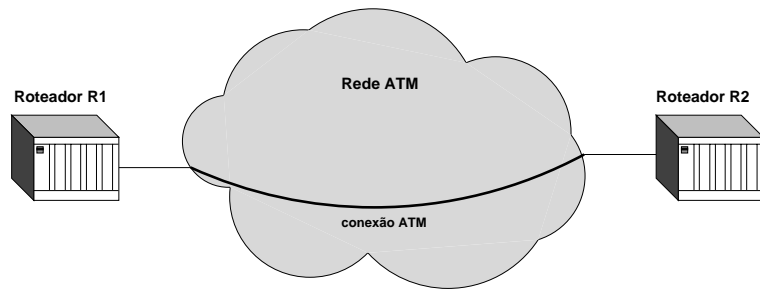


Figura 1.3: Roteadores IP interconectados através de rede ATM

velocidade até redes corporativas de grandes dimensões. É de especial interesse neste texto a interconexão de redes via ATM onde a nuvem ATM serve principalmente para a distribuição de tráfego de outras redes (ATM ou não). Para fins de sistematização, surgiram modelos de interconexão, modelos estes úteis para o entendimento das estratégias de interconexão.

Estes modelos podem ser classificados em dois tipos principais: modelo *Overlay* e modelo *Peer*, e diferenciam-se na forma como a nuvem ATM é vista pelos protocolos existentes, em especial, os protocolos da camada de rede (IP, IPX, etc.). Estes protocolos possuem uma estrutura de endereçamento própria e protocolos de roteamento associados.

Uma possibilidade no relacionamento dos protocolos da camada de rede com a rede ATM consiste numa visão opaca da rede ATM por parte dos protocolos da camada de rede. Esta visão opaca introduz o conceito de nuvem ATM, no qual a camada de rede não “enxerga” o interior da nuvem (figura 1.4). Esta visão significa que a nuvem ATM é vista pelos nós da borda (estações ou roteadores) como uma estrutura de enlace através da qual todos os nós de borda são vizinhos entre si, ou seja, cada um encontra-se à distância de um *hop* um do outro. O fato da nuvem ATM ser constituída de uma estrutura comutada é totalmente invisível pelos nós de borda, ou seja, pelos protocolos da camada de rede destes nós. Esta forma de organizar a interação entre os protocolos de rede e uma rede de transporte é denominada de modelo *Overlay* (ou Modelo Sobreposto).

A interconexão de redes através de uma rede ATM baseada no modelo *Overlay* implica na definição de uma estrutura de endereçamento própria, de protocolos de roteamento e sinalização para a rede ATM. Neste caso, temos a coexistência de duas estruturas de endereçamento diferentes e, consequentemente, protocolos de roteamento e sinalização também diferentes entre as

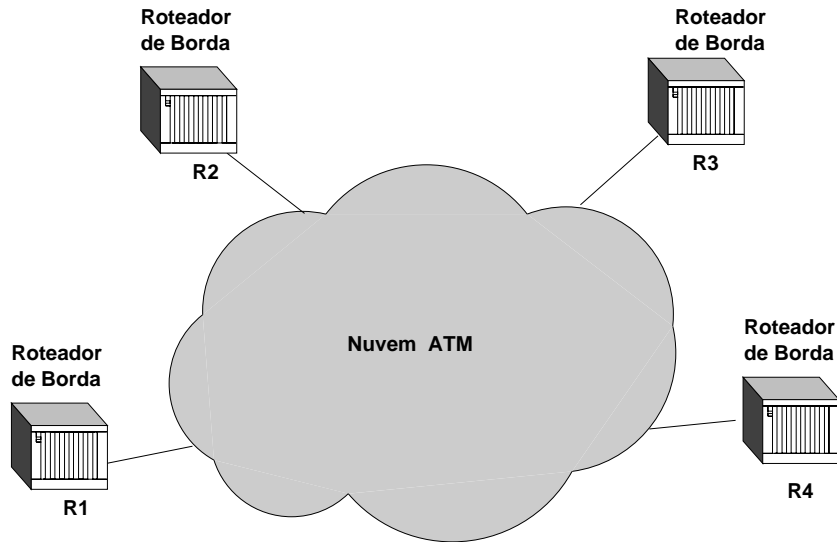


Figura 1.4: Roteadores IP interconectados por uma nuvem ATM

camadas de rede e interface de rede da arquitetura TCP/IP. A existência de duas estruturas diferentes de endereçamento requer um mapeamento do endereço da camada de rede (ex. endereço IP) em um endereço de rede ATM (ex. endereço E.164 ou NSAP). Por exemplo, no caso da figura 1.5, o roteador  $R_1$  recebe um datagrama IP e, após consultar a sua tabela de roteamento descobre que, do ponto de vista do roteamento IP, o próximo *hop* na direção do destino do pacote é o roteador  $R_4$ . Para atravessar a nuvem ATM é necessário descobrir o endereço ATM da interface do roteador  $R_4$ . Após a descoberta do endereço ATM, o datagrama é encapsulado na camada de adaptação ATM e uma conexão é estabelecida (SVC) entre  $R_1$  e  $R_4$ . Para o estabelecimento desta conexão, a sinalização ATM e o roteamento ATM, este último baseado no endereço ATM do roteador  $R_4$ , irão estabelecer o VC (*Virtual Channel*). Todo este procedimento, no nível da nuvem ATM, é completamente desacoplado do roteamento IP. O único acoplamento ocorre quando do mapeamento do endereço IP no endereço ATM respectivo.

A grande vantagem do modelo *Overlay* consiste na independência entre os protocolos que utilizam a rede ATM e os protocolos desta última o que, do ponto de vista da engenharia de protocolos, é importante com relação à independência no desenvolvimento de protocolos. Desta maneira, a tecnologia ATM pode ser utilizada pelos provedores de serviço como um

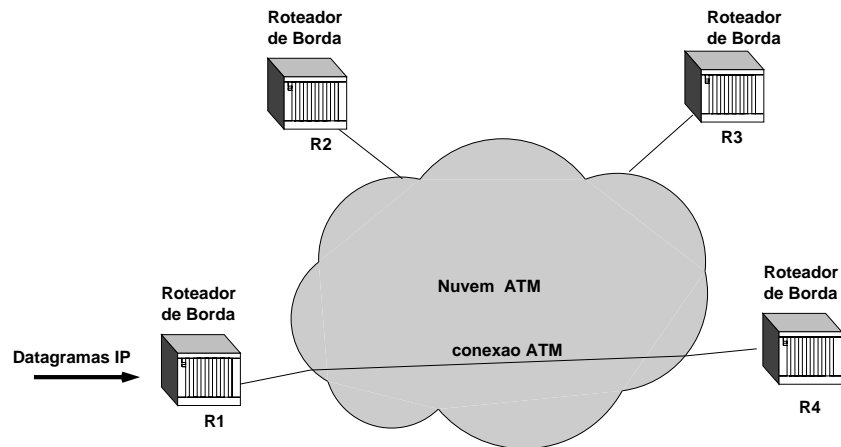


Figura 1.5: Mapeamento do endereço IP de destino no endereço ATM mais próximo

transporte multiserviço, já que a nuvem ATM não fica vinculada a qualquer protocolo da camada superior. O esforço, tanto do IETF como do ATM Forum, consiste em privilegiar o modelo *Overlay* através das soluções IP-Clássico, Emulação de LAN, MPOA, Frame-Relay e SMDS sobre ATM. Como consequência do modelo *Overlay*, o ATM Forum propôs uma estrutura de endereço, no caso de redes privadas, baseada no NSAP (Network Service Access Point) da ISO, e a definição de um protocolo para a NNI (Node-to-Node Interface) denominado PNNI (Private-NNI), constituído de um protocolo de sinalização e um protocolo para roteamento da requisição de sinalização através da rede ATM. No caso das redes públicas é empregado o endereço E.164 e na interconexão através da NNI utiliza-se uma pilha de protocolos baseada na sinalização ITU-T B-ISUP e no protocolo de roteamento nível 3 ITU-T MTP. Estes protocolos são definidos na especificação B-ICI (*Broadband Inter-Carrier Interface*) do ATM Fórum.

A outra forma de interconectar redes através de uma infra-estrutura ATM é quando os protocolos da camada de rede, na borda da infra-estrutura ATM, não têm mais uma visão opaca na forma de nuvem como no modelo *Overlay*. Neste caso, a solução consiste na utilização, dentro da rede ATM, da mesma estrutura de endereço e protocolos de roteamento da camada de rede, o que torna os comutadores ATM em roteadores IP. Desta maneira,  $R_1$  e  $R_4$ , no caso da figura 1.6, não são mais vizinhos como no modelo *Overlay*. Nesta situação, os equipamentos no interior da rede ATM podem realizar o

roteamento convencional de pacote, como também, podem, segundo algum critério, comutar os pacotes diretamente (*forwarding*) no nível do hardware ATM sem que estes pacotes sejam roteados na forma padrão. Esta forma de operar denomina-se de modelo *Peer*.

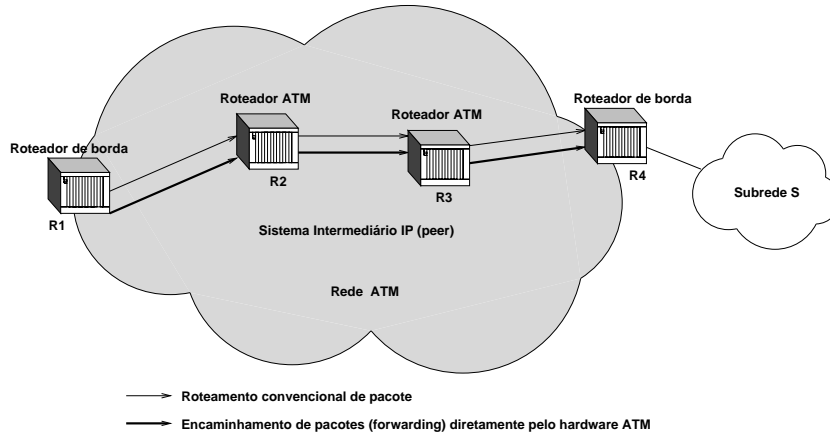


Figura 1.6: Modelo Peer

### 1.3.1 O Modelo Overlay

Conforme discutido anteriormente, o modelo *Overlay* caracteriza-se pela utilização de estruturas de endereçamento diferentes entre os protocolos superiores e a infra-estrutura ATM. A camada de rede com protocolos tais como IP, IPX, AppleTalk e DECnet possuem estruturas de endereçamento próprias. No caso do endereço IP, por exemplo, cada interface possui um endereço único do ponto de vista global cuja atribuição é controlada por uma entidade responsável pela atribuição de endereços. Esta característica de endereço único é a base dos protocolos de roteamento utilizados para o protocolo IP. No caso do ATM, o ATM Forum especificou o protocolo PNNI onde o roteamento entre comutadores ATM é uma das funções principais. Conseqüentemente, o protocolo utiliza muitas das características dos protocolos de roteamento do nível de rede, em especial, dos protocolos baseados no estado do enlace como o OSPF (*Open Short Path First*) e IS-IS (*Intermediate-System to Intermediate-System Routing*).

A forma como o modelo *Overlay* integra a nuvem ATM com os protocolos atualmente em operação permite visualizar a rede de transporte ATM como equivalente a uma estrutura do nível de enlace como, por exemplo, uma

rede Ethernet. No caso desta última, o endereçamento da camada de enlace é baseado em endereços atribuídos à interface física. Em muitos sistemas este endereço corresponde ao endereço MAC (*Medium Access Control*) de 48 bits e implementado na interface de rede pelo fabricante a partir de um endereço administrado pelo IEEE. Conseqüentemente, o projetista de rede não possui controle sobre a alocação do endereço MAC<sup>1</sup>. Neste caso, um dos problemas básicos relativamente aos protocolos superiores, em especial dos protocolos de nível de rede, consiste no mapeamento do endereço de rede em um endereço do nível de enlace como forma de viabilizar a entrega da informação ao destinatário.

A resolução em um endereço nível 2 (nível de enlace) dado um endereço nível 3 (nível de rede) é um processo realizado nas redes com facilidade de difusão através do protocolo ARP (*Address Resolution Protocol*). Em uma rede ATM a situação é um pouco mais complexa pelo fato desta ser uma rede NBMA, ou seja, uma rede sem mecanismos naturais de difusão. Neste caso, é necessário o emprego de servidores para resolução de endereços como teremos oportunidade de discutir na apresentação das propostas mais importantes que empregam o modelo *Overlay*.

### 1.3.2 O Modelo Peer

O modelo *Peer*, contrariamente ao modelo *Overlay*, procura integrar a tecnologia ATM e os protocolos superiores (IP, IPX, etc.). O modelo *Peer* trata os comutadores ATM como elementos de rede (*peers*) dotando-os de certas funcionalidades da camada 3, principalmente roteamento no nível de rede. O modelo *Peer* tem como motivação principal a eliminação da duplicidade de funções, notadamente as funções de roteamento. Existem algumas estratégias para promover essa integração:

1. substituir o endereçamento, roteamento e sinalização padronizados pelo ATM Forum por padrões associados a determinado protocolo de rede (IP, por exemplo);
2. mapear, via procedimento algorítmico, o endereço de rede (IP, por exemplo) em endereço ATM (NSAP ou E.164);
3. empregar um protocolo de roteamento integrado, utilizado tanto pelos protocolos de roteamento no nível de rede quanto pela sinalização ATM. Esta é a proposta do PNNI integrado (I-PNNI).

---

<sup>1</sup>Exceção à esta regra é a rede DECnet que permite a redefinição de endereços MAC.

A primeira solução é adotada em produtos comerciais como no IP Switching da Ipsilon Networks, TAG Switching da Cisco Systems, e implementações MPLS. Esta última solução será explorada neste texto no capítulo 2.

A segunda opção, apesar de conceitualmente interessante, ainda não se mostrou factível em termos de soluções implementadas. A terceira opção tem como ponto favorável o fato do protocolo PNNI possuir funcionalidades equivalentes a protocolos de roteamento atuais como o OSPF, além de incorporar qualidade de serviço ao roteamento. Isto permitiria sua imediata extensão para operar no nível de rede. Como desvantagem, demandaria a substituição de protocolos de roteamento bem difundidos na comunidade de redes de computadores por outro bastante recente. A aceitação desta solução também é incerta.

## 1.4 Considerações Sobre Roteamento

O protocolo IP, por operar sem conexão, exige dos roteadores uma complexa ação de roteamento para cada pacote em trânsito. Esta ação de roteamento é tão mais penosa quanto mais próximo do *backbone* principal da rede o roteador se encontra. Por exemplo, no caso da Internet, um roteador núcleo armazena tabelas de roteamento com cerca de 50.000 entradas. Assim sendo, quando o datagrama trafega por várias subredes o roteamento causa um atraso fim-a-fim intolerável para uma vasta gama de novas aplicações interativas, principalmente aquelas que manipulam áudio e vídeo em tempo real (exemplo: video-conferência) ou que manipulam grandes volumes de informação (exemplo: WWW).

O roteamento em redes IP é processado em função unicamente no endereço de destino do datagrama. Com base neste endereço o roteador determina o próximo *hop*, isto é, o roteador conectado a uma subrede “mais próxima” da subrede destino.

A situação do roteamento na Internet é crítica atualmente pela pressão exercida pelas altas taxas de transmissão nos enlaces físicos. Com as taxas de transmissão aumentando cada vez mais, os roteadores tendem a tornar-se um gargalo comprometendo as novas aplicações. As duas formas básicas para solução deste problema consistem 1) na substituição da malha de roteadores no *backbone* central por uma malha de chaves (*switches*) ATM, e 2) na utilização de *gigarouters*. Esta versão do relatório irá discutir, basicamente, a solução do tipo 1. Os *gigarouters* serão abordados na próxima edição do relatório.

### 1.4.1 O roteamento IP e o Modelo *Overlay*

A necessidade de diminuir o atraso na comutação de pacotes levou à substituição da malha de roteadores no *backbone* central por uma malha de chaves ATM (a nuvem ATM). Seja uma configuração tradicional onde uma nuvem ATM conecta roteadores (denominados roteadores de borda, figura 1.7). Nesta topologia, supondo uma solução segundo o modelo *Overlay*:

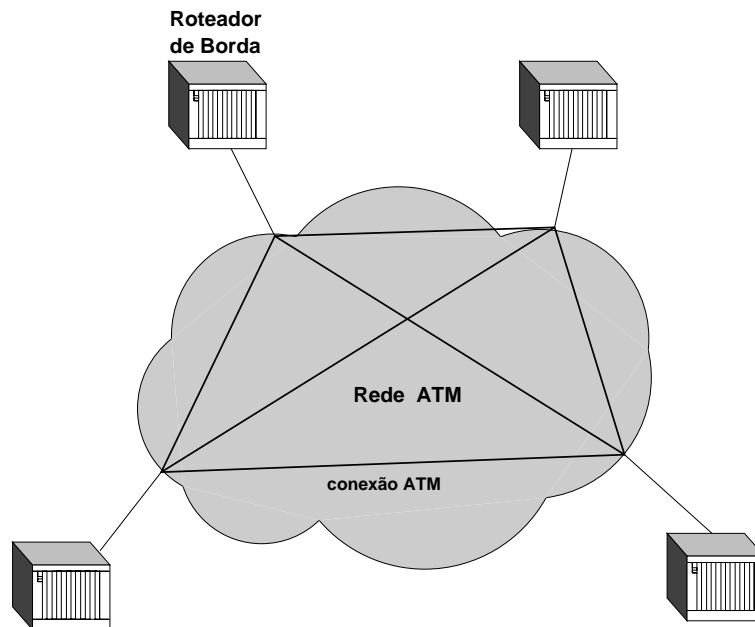


Figura 1.7: Nuvem ATM como malha de interconexão de roteadores

- cada roteador está à distância lógica de um hop de qualquer outro roteador (isto é, cada roteador mantém uma conexão ATM com os demais);
- cada roteador mantém  $N(N-1)/2$  conexões para se interligar aos demais roteadores, onde  $N$  corresponde ao total de roteadores conectados à nuvem;
- a nuvem ATM se apresenta como uma infra-estrutura de camada 2 para os protocolos de rede.

Para redes comutadas de vastas dimensões (como as WANs) a estrutura acima apresenta pelo menos três problemas relacionados ao roteamento:



1. o problema das  $N^2$  conexões: o número de conexões que cada roteador de borda deve manter cresce proporcionalmente ao quadrado do número de roteadores conectados à nuvem;
2. o problema da “vizinhança”: cada roteador de borda mantém relação de vizinhança com todos os demais, o que complica a ação dos protocolos de roteamento, aumenta o tamanho das tabelas de roteamento, e aumenta o tráfego de mensagens de roteamento pela rede;
3. o problema da sobreposição de funções de roteamento: o roteamento no nível de rede, processado pelos roteadores de borda, é independente do roteamento de sinalização processado pelas chaves ATM no interior da nuvem. Os protocolos de roteamento nas bordas e interior à nuvem possuem complexidade equivalente (exemplo: OSPF e PNNI).

O problema das  $N^2$  conexões é resolvido através da criação de redes lógicas na nuvem ATM conforme ilustrado na figura 1.8 onde os equipamentos de borda (estações ou roteadores) estão conectados a 4 redes lógicas (RL's) diferentes,  $RL_1$ ,  $RL_2$ ,  $RL_3$  e  $RL_4$ . Deve ser observado que estas redes lógicas são constituídas independentemente da posição física dos dispositivos de borda. Com a introdução destas redes, um roteador é vizinho somente dos roteadores que se encontram nas mesmas redes lógicas nas quais o roteador está conectado. Desta forma, uma estação que se encontra em uma RL e deseja enviar um pacote para uma outra estação conectada à nuvem ATM mas em outra rede lógica, terá que encaminhar o pacote para um dos roteadores na sua rede lógica, e este se encarregará de encaminhar o pacote para o próximo *hop* até alcançar um roteador que se encontre na mesma rede lógica onde está conectada a estação destino.

A organização da nuvem ATM em redes lógicas permite introduzir o conceito do *roteador virtual*. Como visto anteriormente, a comunicação entre estações em redes lógicas diferentes terá os seus pacotes encaminhados *hop-by-hop* através de roteadores conectados às redes lógicas. O que deve ser destacado é que as estações encontram-se conectadas na mesma nuvem ATM o que permitiria uma abertura de conexão direta entre estas estações (atalho). Este conceito é ilustrado na figura 1.9 onde é mostrado o caminho correspondendo ao roteamento *hop-by-hop* envolvendo o tráfego entre os roteadores  $R_1$  e  $R_4$ , e a conexão ligando diretamente os dois roteadores através de uma conexão ATM.

Roteamento virtual pode ser entendido como uma agregação de roteadores num “mega-roteador”. Seja o modelo de roteador convencional da figura 1.10 onde as funções de roteamento são executada pelo processador

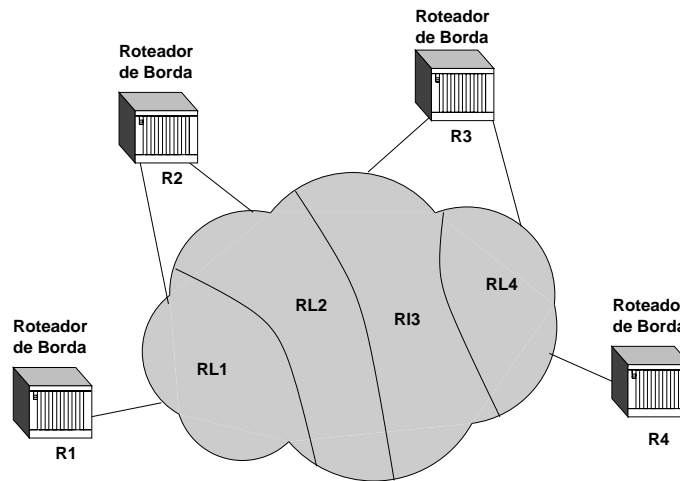


Figura 1.8: Introdução de Redes Lógicas (RL's)

de rede e as funções de encaminhamento são executadas pelas interfaces de rede através do barramento interno. Na figura 1.9 percebe-se que a nuvem ATM faz o papel de barramento que conecta todas as interfaces de rede dos roteadores. As interfaces de rede do roteador virtual da figura 1.10 fazem o papel dos roteadores de borda da figura 1.9, e o processador de roteamento corresponde à funcionalidade do protocolo de roteamento executado entre os roteadores de borda da figura 1.9.

No roteamento virtual, a comutação tradicional de pacotes de roteador para roteador (*hop-by-hop*) cede lugar ao chaveamento de pacotes. O chaveamento de pacotes se processa através do estabelecimento de um atalho (*shortcut*) entre dois pontos na rota do pacote (pontos de ingresso e egresso, respectivamente). No ponto de ingresso o pacote é segmentado em células ATM que são comutadas através do atalho. No ponto de egresso as células são remontadas para a obtenção do pacote original.

O roteamento virtual apresenta certas vantagens quando comparado ao roteamento *hop-by-hop*, por exemplo:

- o baixo atraso de propagação das células ATM através da rede e a baixa taxa de perda de célula resulta num tráfego eficiente de pacotes (propagação com baixa perda e atraso);
- num atalho o pacote é inspecionado apenas na origem e no destino, não em cada comutador intermediário;

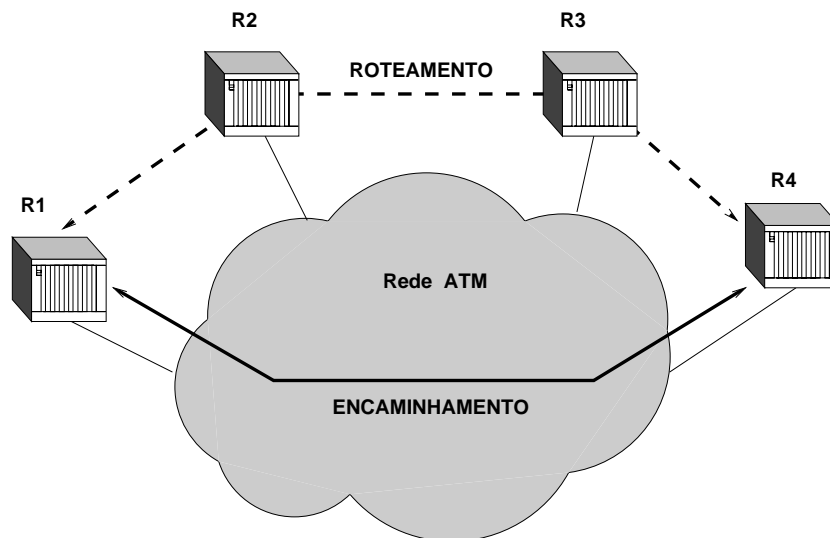


Figura 1.9: Funções de roteamento e encaminhamento numa nuvem ATM

- o tráfego de pacotes através de atalhos libera recursos para o tráfego via roteadores, possibilitando um balanceamento entre os tráfegos *hop-by-hop* e o tráfego representado pela comutação dos pacotes no atalho estabelecido;
- possibilidade de utilização de qualidade de serviço propiciada pela rede ATM entre os roteadores de entrada e saída da nuvem ATM.

Entretanto, a técnica de roteamento virtual demanda políticas eficazes de gerenciamento de atalhos. A decisão sobre o momento de estabelecer e encerrar um atalho é crucial. O *overhead* associado ao seu estabelecimento somente será compensado caso um grande número de pacotes seja comutado através do atalho. Como regra geral, a política de estabelecimento de atalhos pode ser orientada a fluxo ou a topologia. No primeiro caso tem-se o chaveamento de pacotes orientado a fluxo, enquanto no segundo caso tem-se o chaveamento de pacotes orientado a topologia. Estes conceitos serão discutidos nas próximas seções.

#### 1.4.2 O Roteamento IP e o Modelo *Peer*

Ao analisarmos a forma de operação de uma rede de transporte como o ATM baseada no modelo *Overlay*, podemos distinguir que o roteamento

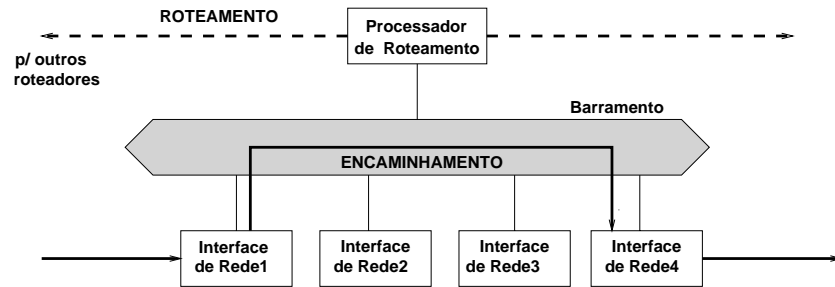


Figura 1.10: Roteamento Virtual

virtual cria dois caminhos distintos: 1) o caminho *hop-by-hop*, associado ao roteamento padrão e 2) o atalho criado conectando 2 elementos na borda da nuvem. Quais pacotes vão percorrer o caminho *hop-by-hop* do roteamento padrão, ou quais pacotes irão ser encaminhados pelo atalho, depende de um critério a ser executado pelo dispositivo de borda na entrada da rede.

No modelo *Peer* a característica essencial é que os elementos que constituem a estrutura comutada da rede ATM são, na realidade, roteadores baseados no hardware ATM. Estes roteadores ATM executam protocolos de camada de rede como OSPF, RIP e o BGP, e não mais as funções de roteamento e sinalização ATM como PNNI.

Da mesma forma como no modelo *Overlay*, o modelo *Peer* irá distinguir dois caminhos para o tráfego dos pacotes: 1) um caminho que corresponderá ao roteamento padrão *hop-by-hop* e um caminho comutado no nível do hardware. No modelo *Peer*, como existe um único protocolo de roteamento, diferentemente do modelo *Overlay* onde existem 2 estruturas de roteamento distintas gerando 2 caminhos diferentes, no modelo *Peer* o caminho correspondente ao roteamento *hop-by-hop* (caminho roteado) e o caminho comutado percorrerão, topologicamente, a mesma rota dentro da rede de transporte ATM. O que acontece neste caso, é que o caminho comutado tende a oferecer uma maior eficiência no transporte de pacotes do que o caminho roteado, especialmente em se tratando de um hardware ATM, isto porque, é necessário recuperar o pacote em cada *hop* através da remontagem das células correspondentes, realizar o roteamento padrão, e segmentar de novo o pacote em células antes de reencaminhá-lo para o próximo *hop*. No caso do caminho comutado, os datagramas são encaminhados baseados diretamente no nível das células tendo como referência o VPI/VCI das células que compõem o pacote ou, mais precisamente, o fluxo de pacotes. Genericamente, este con-

ceito pode ser estendido de modo a que, no modelo *Peer*, os pacotes enviados no caminho comutado sejam encaminhados através de um *label* associado, seja este *label* o par VPI/VCI do ATM, o DLCI do Frame-Relay, ou mesmo um símbolo qualquer que seja encapsulado no cabeçalho de um pacote. Neste último caso, o pacote será encaminhado não mais através do exame do endereço do destino em cada *hop*, mas sim, através de um símbolo que é atribuído na entrada da rede e que permitirá que ele atravesse a rede de transporte sem sofrer roteamento. A figura 1.11 ilustra os caminhos roteado e comutado no modelo *Peer*. Note que esta figura é um detalhamento da figura 1.6.

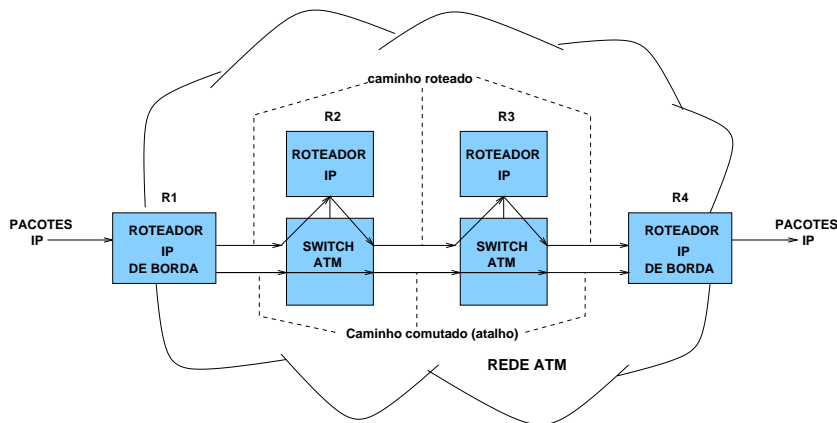


Figura 1.11: Modelo Peer: caminho roteado x caminho comutado

## 1.5 Chaveamento de Pacotes Orientado a Fluxo

Fluxo é uma sequência de pacotes de uma ou mais fonte emissora para um determinado destino (*unicast* ou *multicast*). Um fluxo é sempre unidirecional. São exemplos de fluxo:

- o *download* de um documento WWW;
- a transferência de um arquivo longo via protocolo FTP;
- a transmissão de áudio ou vídeo através do MBONE;
- uma sessão TELNET.

A importância do conceito de fluxo é tal que o protocolo IP versão 6 (IPv6) reserva um campo de 24 bits para associar um datagrama a um fluxo. A utilização deste campo ainda não está plenamente padronizada para o IPv6.

Fluxos podem ser caracterizados de acordo com as seguintes propriedades:

- fluxos definidos pelo destino: todos os datagramas IP cujos endereços IP de destino coincidem pertencem a um mesmo fluxo;
- fluxos definidos pela origem e destino: todos os datagramas cujos endereços IP de origem e de destino coincidem pertencem ao mesmo fluxo;
- fluxos definidos por aplicações: todos os datagramas IP cujos seguintes campos coincidem:
  - endereços IP de origem e de destino;
  - protocolo de transporte;
  - *ports* de origem e destino (definidos no cabeçalho dos protocolos de transporte TCP ou UDP).

As formas acima definem a granularidade do fluxo. Por exemplo pode-se caracterizar o fluxo definindo apenas um endereço de destino; um endereço de origem e um de destino; ou um par de aplicações identificadas pela tripla (endereço, protocolo de transport, *port*).

Um fluxo pode ser caracterizado pela fonte emissora, por exemplo, associando um mesmo identificador no campo correspondente do IPv6 para todos os datagramas pertencentes ao fluxo. Dado que a arquitetura TCP/IP ainda utiliza a versão 4 do protocolo IP (IPv4) que não possui campo para caracterizar fluxo, outras formas de identificação devem ser exploradas. A identificação pode se basear na detecção (automática) do fluxo ou via ação de gerência.

Chaveamento de pacotes orientado a fluxo estabelece um atalho após a detecção de um fluxo. Os  $N$  pacotes necessários para caracterizar o fluxo são comutados *hop-by-hop* sendo os demais chaveados através do atalho após o seu estabelecimento. A figura 1.12 ilustra esta estratégia.

Supondo uma interconexão segundo o modelo *Overlay* o estabelecimento de um atalho deve seguir a seguinte ordem temporal:

1. um fluxo é detectado num equipamento de borda;

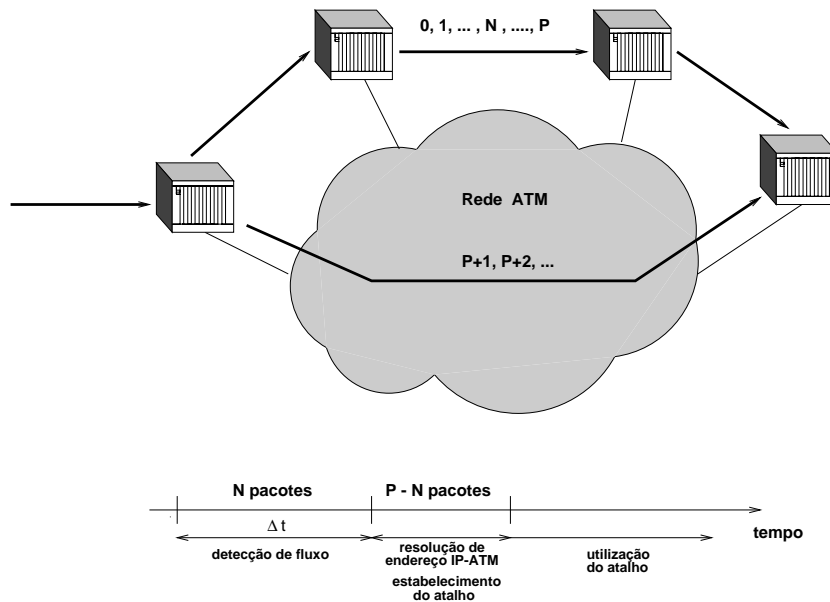


Figura 1.12: Chaveamento de pacotes orientado à fluxo

2. o equipamento de borda inicia um processo de descoberta do endereço ATM do equipamento de egresso (enquanto isso não se completa, os pacotes do fluxo continuam sendo roteados de maneira convencional);
3. descoberto o endereço ATM de egresso o equipamento de ingresso estabelece uma conexão ATM (atalho) com o equipamento de egresso utilizando sinalização e roteamento ATM;
4. estabelecida a conexão, os pacotes pertencentes ao fluxo são enviados através da conexão utilizando um encapsulamento pré-estabelecido ou negociado;
5. cessando o fluxo, a conexão é terminada após um período de inatividade.

O protocolo MPOA (*Multiprotocol Over ATM*) descrito no capítulo ?? opera exatamente desta forma.

O conceito de fluxo também é importante no caso do modelo *Peer*. A partir do instante que um roteador-ATM identifica, segundo algum critério, um determinado fluxo de pacotes, o roteador-ATM pode distribuir um *label* para os pacotes pertencentes ao fluxo (no caso do ATM o *label* traduz-se, por

exemplo, no par VPI/VCI) e distribui, através de um protocolo específico, o *label* para os roteadores-ATM que se encontram na direção oposta ao fluxo de pacotes. O objetivo é que a partir deste instante, os roteadores-ATM que recebem o *label* (VPI/VCI) irão comutar os pacotes, que fazem parte do fluxo, diretamente no nível de célula utilizando o *label* (VPI/VCI) recebido para identificar o fluxo. Este mesmo comportamento por parte de todos os roteadores-ATM irá permitir a construção de um caminho comutado entre o dispositivo de borda na entrada da rede e o dispositivo de borda na saída da rede. Este caminho pode ser visto, comparativamente ao caminho roteado, como sendo também um atalho (*shortcut*).

## 1.6 Chaveamento de Pacotes Orientado a Topologia

No chaveamento de pacotes orientado a topologia, o ponto terminal de um atalho é função da topologia da rede, topologia esta determinada pelos protocolos de roteamento convencionais. Em linhas gerais, o chaveamento de pacotes orientado a topologia constrói atalhos com base nas informações de controle (roteamento), enquanto o chaveamento orientado a fluxo constrói atalhos com base no fluxo de dados. A primeira estratégia é denominada *orientada por controle* (*control-driven*), enquanto a segunda é denominada *orientada por dados* (*data-driven*).

O chaveamento orientado a topologia é mais escalável que o chaveamento orientado a fluxo por razões explicitadas a seguir:

1. Capacidade de agregação: no chaveamento orientado a topologia os fluxos que convergem a um ponto de egresso são agregados num único atalho. Esta característica confere maior escalabilidade ao chaveamento orientado a topologia pois o número de atalhos é bem menor que o número de fluxos.
2. Menor sobrecarga de sinalização: no chaveamento orientado a fluxo um circuito virtual é estabelecido e terminado segundo a dinâmica de fluxos. Quando orientado a topologia, atalhos são estabelecidos segundo a dinâmica do roteamento, dinâmica esta muito mais “lenta” que a dinâmica das aplicações geradoras de fluxos.
3. Crescimento dependente do tamanho da rede: no chaveamento orientado a topologia o número de atalhos cresce em função do número de



## 1.6. CHAVEAMENTO DE PACOTES ORIENTADO A TOPOLOGIA 25

rotas, enquanto no chaveamento orientado a fluxo o número de conexões é função do número de aplicações produtoras/consumidoras de fluxo que se comunicam através da nuvem ATM.

4. Todos os pacotes num atalho são chaveados (lembre-se que o chaveamento orientado a fluxo comuta os pacotes após a detecção do fluxo e estabelecimento do atalho).
5. Possibilidade de estabelecimento de atalhos via ação de gerência, independente de roteamento. Esta característica permite a implantação de políticas de engenharia de tráfego nas infra-estruturas ATM com chaveamento orientado a topologia.

Entretanto, o chaveamento orientado a topologia apresenta também certas desvantagens. Atalhos são criados e permanecem ativos mesmo quando inexistente tráfego de dados que possa se beneficiar do atalho. Outro problema diz respeito à junção de tráfego proveniente de fontes distintas. Como a AAL5 é incapaz de multiplexar tráfego, soluções externas à comutação/adaptação ATM devem ser elaboradas<sup>2</sup>.

Um cenário de utilização do chaveamento orientado a topologia baseado modelo *Peer* é mostrado a seguir:

1. um dispositivo de borda de saída descobre uma rota para determinada subrede;
2. o dispositivo de borda decide criar um atalho associado à rota;
3. com um protocolo específico, o dispositivo de borda distribui esta sua decisão para os dispositivos intermediários no interior da nuvem;
4. os dispositivos intermediários registram este atalho, atribuindo valores VPI/VCI em cada *link* que compõe o atalho;
5. ao receber um pacote, um dispositivo de ingresso examina se existe um atalho associado àquela rota. Em existindo, o pacote é comutado através do atalho.

A figura 1.13 ilustra este procedimento. O protocolo MPLS (*Multi-Protocol Label Switching*) descrito no capítulo 2 opera exatamente desta forma.

---

<sup>2</sup>Estas soluções serão discutidas no âmbito do MPLS (capítulo 2).

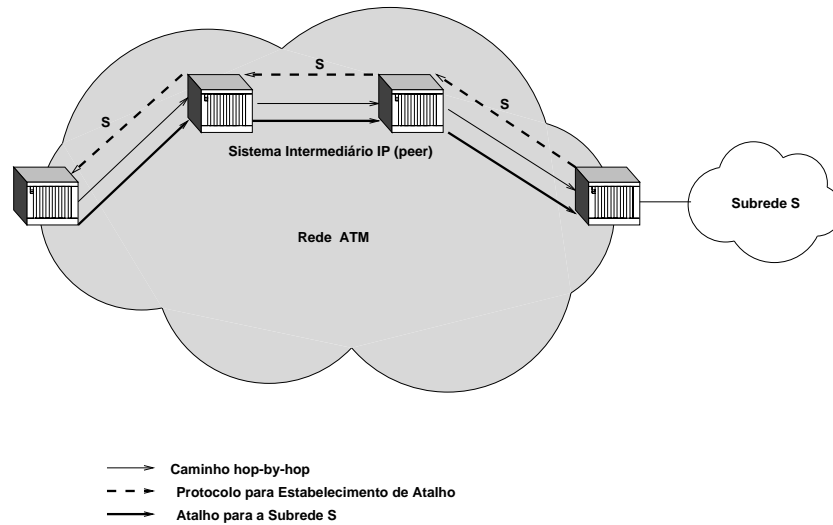


Figura 1.13: Chaveamento de pacotes orientado à topologia

Como conclusão deste capítulo podemos afirmar que a diferença fundamental entre os modelos *Overlay* e *Peer* pode ser visualizada na figura 1.14. Ambos os modelos procuram estabelecer atalhos a fim de explorar a velocidade e confiabilidade da comutação ATM. Via de regra, no modelo *Overlay* necessita-se de um protocolo para mapear endereços IP em endereços ATM. Neste modelo, atalhos são estabelecidos via sinalização e roteamento ATM. *O tráfego hop-by-hop não necessariamente possui a mesma rota do tráfego chaveado pois os protocolos de roteamento IP e ATM são distintos.*

O modelo *Peer* dispensa sinalização e roteamento ATM mas demanda um protocolo para a criação de atalhos. Este protocolo utiliza as mesmas rotas estabelecidas pelos protocolos de roteamento. Como consequência, *o tráfego chaveado possui a mesma rota do tráfego hop-by-hop.*

1.6. CHAVEAMENTO DE PACOTES ORIENTADO A TOPOLOGIA 27

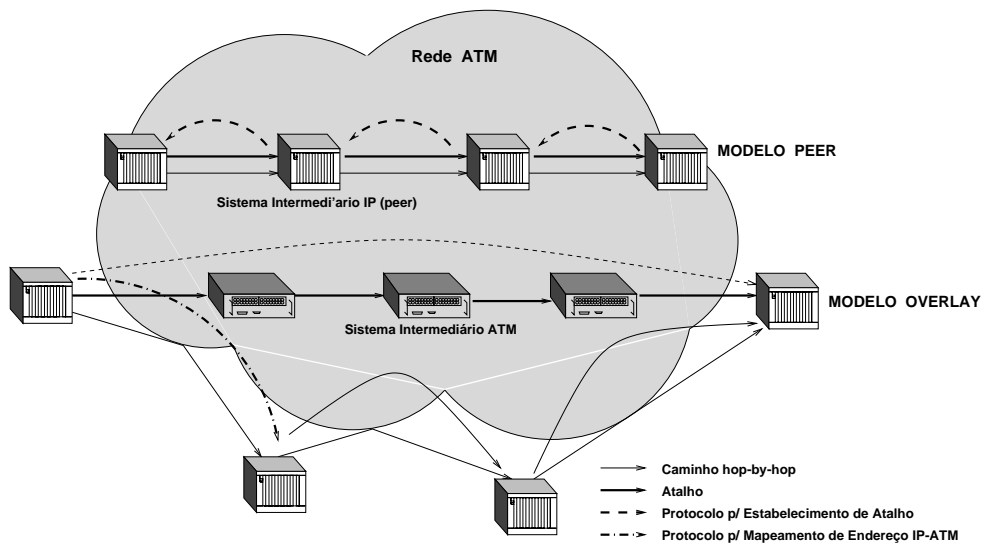


Figura 1.14: Diferença fundamental entre os modelos *Peer* e *Overlay*



## Capítulo 2

# MPLS - Multiprotocol Label Switching

O surgimento nos últimos anos de tecnologias como o *IP-Switching*, *TAG-Switching*, *Cell Relay Switching (CSR)*, *ARIS (Aggregate Route-Based IP Switching)* etc., todas pertencentes ao modelo *Peer*, indicavam o interesse da indústria no desenvolvimento de novas tecnologias para a comutação IP, em detrimento do modelo *Overlay* adotado pelo ATM Fórum e o IETF através das respectivas soluções baseadas no MPOA (*Multiprotocol Over ATM*) e no IP-Clássico.

Em dezembro de 1996 foi decidida a formação de um grupo de trabalho no âmbito do IETF que teria como foco a integração do roteamento e da comutação através da definição de um padrão que englobasse os principais aspectos das tecnologias baseadas no modelo *Peer*. Esta decisão originou o grupo de trabalho MPLS que surgiu com o objetivo de integrar o roteamento, no nível da camada de rede, e a comutação por rótulos em uma única solução, procurando atender os seguintes objetivos:

- melhoria nos custos e no desempenho do roteamento;
- melhoria na escalabilidade do roteamento relativamente ao modelo *Overlay*;
- maior flexibilidade na introdução de novos serviços.

O último objetivo é de grande interesse para os provedores de serviços de rede que percebem a necessidade de oferecer, no futuro próximo, novos serviços como forma de atrair mais usuários e de aumentar suas receitas.

## 2.1 Características Gerais

Uma infra-estrutura MPLS é capaz de encaminhar pacotes no nível 3 através do processamento padrão de cada pacote e de encaminhar pacotes no nível 2 através de uma técnica baseada na comutação de rótulos (*labels*). Como membro do modelo *Peer*, cada dispositivo MPLS executa um único protocolo de roteamento IP. Neste caso, diferentemente do modelo *Overlay*, é mantida uma topologia única e um espaço único de endereço. A operação básica do MPLS consiste no mapeamento de *labels* a *Classes Equivalentes de Encaminhamento (FECs: Forwarding Equivalent Class)*. Uma *FEC* corresponde a um grupo de pacotes IP que são encaminhados da mesma maneira, isto é, percorrem a mesma rota e recebem o mesmo tratamento no seu encaminhamento. Para viabilizar este mapeamento é definido um *Protocolo de Distribuição de Labels (LDP: Label Distribution Protocol)* cuja função é distribuir as associações *Label/FEC (L/F)* entre os dispositivos MPLS. Estes últimos são denominados de *LSRs (Label Switch Routers)*. Após a distribuição dos *labels* pelo LDP, tem-se o estabelecimento de um *LSP (Label Switching Path)* que liga um *LSR* de ingresso na rede a um *LSR* de egresso da rede conforme ilustrado na figura 2.1.

Quando da chegada de um pacote à rede, o *LSR* de ingresso examina os vários campos do cabeçalho do pacote e determina a *FEC* à qual o pacote corresponde. Caso exista uma associação de *label* para esta *FEC* e, conseqüentemente, um *LSP* estabelecido, o *LSR* de entrada introduz o *label* no pacote e envia-o para a interface de saída correspondente. A partir deste instante, o pacote é comutado pelo *label* através da rede até alcançar o *LSR* de saída onde o *label* é removido e o pacote é processado no nível 3. O ganho de desempenho proporcionado pelo MPLS deve-se ao fato de que o processamento de nível 3 é realizado somente uma vez, isto é, na entrada da rede MPLS, ao invés de acontecer em cada *hop* dentro da rede como no encaminhamento IP padrão. O MPLS modifica um paradigma fundamental hoje existente nas redes IP: a superposição de um rótulo ao datagrama tem a propriedade de imprimir à comunicação uma característica de “orientação a conexão”.

Conforme ilustrado na figura 2.2, MPLS desacopla as funções de roteamento IP e de encaminhamento em dois planos distintos. Os comutadores ATM que suportam MPLS comportam-se como roteadores IP. Neste caso, o comutador executa protocolos de roteamento do tipo OSPF (*Open Shortest Path First*), RIP (*Routing Information Protocol*), BGP (*Border Gateway Protocol*), etc., e possuem endereços IP, não mais endereços ATM.

Os endereços IP destes roteadores ATM são anunciados pelos protocolos

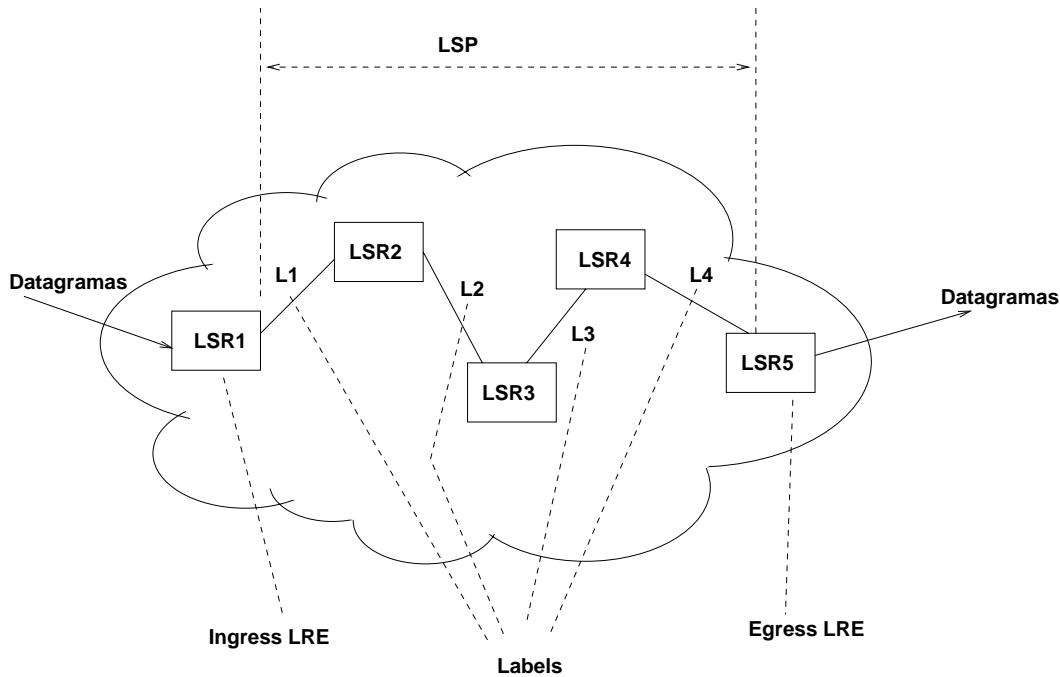


Figura 2.1: Estabelecimento de um LSP (*Label Switched Path*)

de roteamento da mesma forma quando são utilizados roteadores convencionais. Os datagramas correspondentes aos protocolos do roteamento são mapeados na AAL5 e distribuídos nas interfaces ATM em um VC (*Virtual Channel*) reservado especificamente para este fim. Nos roteadores convencionais a função de roteamento IP já é parte integrante destes dispositivos.

Em ambos os casos, isto é, roteadores ATM (*LSRs – ATM*) e roteadores convencionais, o roteamento no nível da camada de rede mantém informações dos protocolos OSPF e BGP para determinar como os pacotes devem ser roteados. A partir destas informações é possível particionar o espaço de encaminhamento de modo a criar FECs correspondendo, em geral, aos prefixos de rede armazenados nas entradas da tabela de roteamento.

Para cada FEC é atribuído um rótulo pequeno, de comprimento fixo e de significado local a um dado ponto de encaminhamento. No MPLS, a atribuição de um determinado pacote a uma FEC é feita apenas uma vez, quando o pacote entra na rede MPLS. Ao entrar na rede, além de se fazer a análise convencional de encaminhamento de camada 3 em busca do próximo nó, faz-se também a busca para descobrir a FEC à qual o pacote pertence e

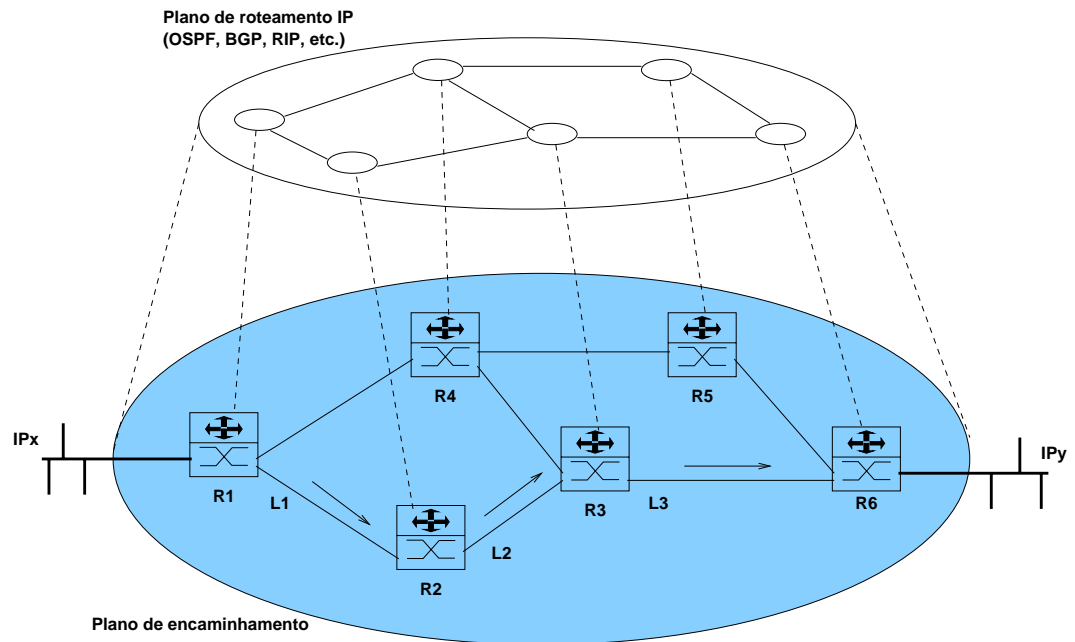


Figura 2.2: Desacoplamento dos Planos de Roteamento e Encaminhamento no MPLS

o respectivo rótulo associado. Quando o pacote é enviado ao próximo nó, o rótulo é enviado junto. Em roteadores convencionais suportando MPLS, um pacote é rotulado codificando-se um rótulo entre seus cabeçalhos de nível 2 e nível 3. Quando se emprega comutadores ATM como roteadores IP baseados em MPLS, o rótulo é mapeado nos campos VPI/VCI da célula ATM.

Nos nós subsequentes, não há mais análise do cabeçalho de camada de rede do pacote. Ao invés disso, o rótulo é usado como um índice para uma tabela a qual especifica o próximo nó e um novo rótulo. O rótulo velho é substituído pelo novo rótulo, e o pacote é encaminhado a seu próximo nó. Se a atribuição a uma FEC for baseada num "casamento mais longo" de prefixo, por exemplo, isto elimina a necessidade de realizar o cálculo de "casamento mais longo" para cada pacote em cada roteador. Este cálculo é realizado apenas uma vez quando do estabelecimento do LSP.

Na figura 2.2 anterior, um nó com endereço  $IP_x$  desejando enviar pacotes para um nó de endereço  $IP_y$ , coloca os pacotes no meio compartilhado de sua LAN com o respectivo endereço destino  $IP_y$ . O roteador  $R_1$  faz a análise e determina em sua tabela de encaminhamento que o destino  $IP_y$  corresponde



a um "casamento mais longo" com um prefixo associado a uma FEC  $\mathcal{F}_\dagger$ , tendo como próximo hop o roteador  $R_2$  e para a qual  $R_2$  distribuiu um rótulo L1. Este pacote é rotulado com L1 e encaminhado na interface de saída que dá acesso a  $R_2$ . Em  $R_2$  não é mais necessário fazer a análise de prefixo. Basta  $R_2$  consultar sua tabela de rótulos através de uma indexação simples e verificar a qual interface o pacote deverá ser encaminhado e com qual rótulo. Supõe-se que para  $R_2$  tenha sido distribuído um rótulo L2 para a mesma FEC  $\mathcal{F}_\dagger$ . Então,  $R_2$  substituirá L1 por L2 e encaminhará o pacote a  $R_3$ . O processo se repete nos demais roteadores, até que o pacote atinja o roteador destino, sendo finalmente entregue na LAN onde se encontra o nó com endereço  $IP_y$ . Neste exemplo,  $R_1$  e  $R_6$  são denominados dispositivos de borda, uma vez que fazem a interface entre domínios não MPLS e o domínio MPLS.

O plano de encaminhamento, portanto, é baseado fundamentalmente em dois mecanismos:

1. busca em tabelas de rótulos, as quais definem o encaminhamento a ser dado ao datagrama;
2. substituição do rótulo de entrada por um rótulo de saída e o encaminhamento do datagrama pela interface de saída.

Uma vez feita a distribuição dos rótulos para uma dada  $FEC$ , este mecanismo de encaminhamento é completamente desacoplado do plano de roteamento.

Comutadores ATM suportando MPLS dispensam os mecanismos nativos do ATM, como o PNNI, para estabelecimento de conexão. MPLS define protocolos próprios para a distribuição de rótulos associados a  $FEC's$ , o que substitui os mecanismos nativos do ATM. Estes protocolos são o *Label Distribution Protocol* - LDP e o já conhecido RSVP, com algumas extensões para suportar MPLS. Quando o protocolo LDP (ou RSVP) distribui rótulos associados a uma FEC, na prática ele está estabelecendo conexões através dos comutadores ATM. Estes protocolos serão discutidos nas próximas seções. De forma mais genérica, pode-se dizer que a distribuição de rótulos associados a uma FEC tornará o encaminhamento orientado a conexão, rompendo um paradigma fundamental das redes IP hoje existentes.

MPLS permite inferir a precedência ou classe de serviço a partir do rótulo de forma que não é necessária nenhuma análise adicional do cabeçalho IP. Em certos casos, MPLS provê uma maneira de codificar explicitamente a classe de serviço nos campos do rótulo.

Além do ganho no desempenho, o maior interesse associado ao MPLS, do ponto de vista dos grande operadores de redes, é a possibilidade de realização de funções avançadas de engenharia de tráfego. Estas funções baseiam-se na criação de rotas explícitas diferentes da rota definida pelo roteamento IP. A criação de rotas explícitas permite distribuir o tráfego de diferentes *FECs* através de diferentes caminhos de modo a balancear a carga que atravessa a rede. Outra facilidade viabilizada pelo MPLS é a criação de redes privadas virtuais (*VPNs - Virtual Private Networks*) através do mapeamento do tráfego intra-VPN a *labels* o que permite criar *LSPs* entre domínios de uma mesma VPN.

Uma das características do MPLS é que este pode ser implementado sobre qualquer tecnologia de enlace. Dentre as várias tecnologias de enlace o ATM representa aquela de maior interesse. Neste caso, MPLS deve ser visto como um outro plano de controle dirigido pelo roteamento IP, onde a criação de *LSP* corresponde ao estabelecimento de um VC-ATM. A integração do MPLS com o ATM vem atender o interesse dos provedores de serviços de rede que possuem uma infra-estrutura ATM, no sentido de permitir uma solução que ofereça um melhor uso da infra-estrutura para o transporte de grandes volumes de tráfego IP.

Por outro lado, apesar de inspirada nos mecanismos de comutação de rótulo do ATM, MPLS também possui um grande interesse para os roteadores convencionais. O fato do MPLS introduzir uma característica de orientação a conexão traz benefícios que no roteamento convencional não eram realizáveis. Por exemplo, é possível pensar em empregar recursos de engenharia de tráfego para estes sistemas de forma similar ao que existe disponível para a tecnologia ATM, reduzindo o distanciamento entre estas tecnologias e reduzindo, desta forma, a vantagem do ATM relativamente aos roteadores IP tradicionais. Com o surgimento dos *gigarouters*, capazes de suportar taxas elevadas de comutação de pacotes, o distanciamento entre as tecnologias tende a se reduzir mais ainda.

## 2.2    **Arquitetura MPLS**

No roteamento IP convencional 2 pacotes são considerados pertencentes a uma mesma *FEC*, ou seja, recebem o mesmo tratamento por parte dos roteadores, quando existe uma entrada nas respectivas tabelas de roteamento que corresponda ao prefixo do endereço destino do pacote. À medida que o pacote atravessa a rede, cada roteador examina o pacote e associa-o implicitamente a uma *FEC* na medida em que o pacote recebe o mesmo

tratamento indicado nas tabelas criadas pelo protocolo de roteamento.

Como visto em seção anterior, a associação do pacote a uma *FEC*, no caso do MPLS, é realizada uma única vez, ou seja, na entrada da rede, através da atribuição ao pacote de um *label* de tamanho pequeno e fixo. Nos *hops* subsequentes, o *label* é utilizado como um índice em uma tabela de *labels* a qual especifica o próximo *hop* e um novo *label*. O *label* antigo é substituído pelo novo *label* e o pacote é encaminhado para o próximo *hop*.

Na sequência são apresentadas algumas das vantagens importantes do MPLS com relação ao encaminhamento convencional de pacotes no nível 3:

- o encaminhamento MPLS pode ser efetuado por comutadores capazes de realizar a pesquisa e substituição de *label* mas que não são capazes de analisar o cabeçalho dos pacotes, ou ainda, quando a velocidade de análise do cabeçalho dos pacotes não é suficientemente rápida;
- no momento em que o pacote é associado a uma *FEC*, o *LSR* de entrada pode utilizar qualquer informação que ele possua sobre o pacote, mesmo que esta informação não possa ser inferida a partir do cabeçalho do pacote. Por exemplo, pacotes recebidos em portas diferentes podem ser atribuídos a *FECs* diferentes. No caso do roteamento convencional somente é possível considerar a informação contida no cabeçalho do pacote;
- um pacote que entra na rede através de um determinado roteador pode receber um *label* diferente caso este mesmo pacote entre na rede através de um outro roteador. Desta forma, as decisões de encaminhamento do pacote são dependentes do ponto de entrada na rede. Esta função é realizada de forma trivial no MPLS, diferentemente do roteamento convencional, isto porque a identificação do ponto de entrada do pacote na rede não viaja com o pacote;
- as estratégias de associação de um pacote a uma *FEC* podem tornar-se extremamente sofisticadas sem que isto represente qualquer ônus adicional para o encaminhamento do pacote no interior da rede;
- é possível realizar engenharia de tráfego através do uso do roteamento explícito. O roteamento explícito, no caso do protocolo IP, implica na codificação da rota, ou seja, na indicação explícita dos endereços dos roteadores pertencentes à rota no cabeçalho do pacote (*source routing*). No caso do MPLS, um *label* pode ser utilizado para representar a rota, ou seja, os pacotes não precisam carregar os endereços dos roteadores que devem ser atravessados na rota explícita.

## 2.3 Conceitos Básicos

Na sequência são apresentados os conceitos básicos presentes no MPLS.

### Labels (Rótulos)

Um *label* corresponde a um identificador de tamanho pequeno e fixo, de significado local ao enlace entre dois componentes MPLS (*LSRs*) e que representa uma determinada *FEC*. O pacote que recebe o *label* é tratado como pertencente àquela *FEC*.

Supondo 2 *LSRs*,  $R_u$  ( $R_{upstream}$ ) e  $R_d$  ( $R_{downstream}$ ), o fluxo de pacotes flui de  $R_u$  para  $R_d$  conforme mostrado na figura 2.3, onde  $R_u$  corresponde a  $LSR_{i-1}$  (ou  $LSR_i$ ) e  $R_d$  a  $LSR_i$  (ou  $LSR_{i+1}$ ).

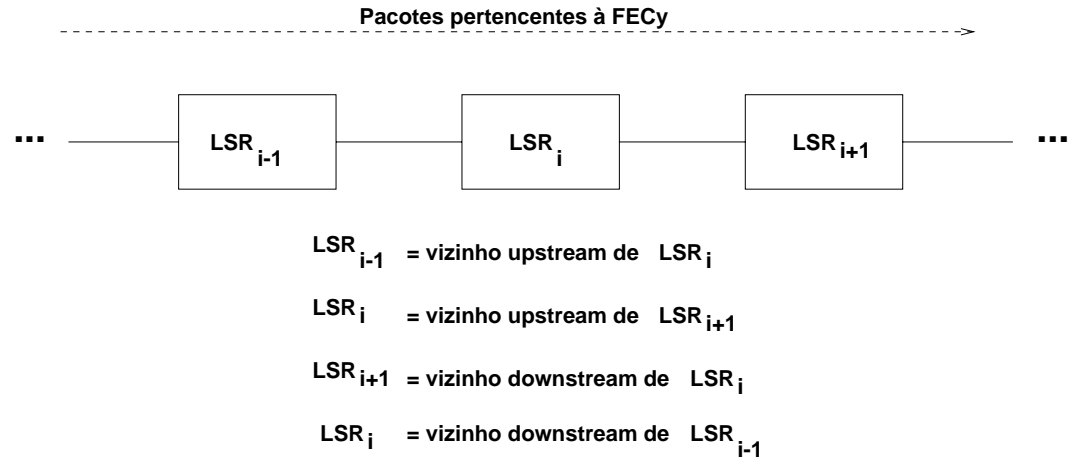


Figura 2.3: Relações de Vizinhança no MPLS

Supondo que os pacotes são membros de uma *FEC* denominada  $\mathcal{F}$ ,  $R_u$  e  $R_d$  concordam que, quando do envio dos pacotes por parte de  $R_u$  para  $R_d$ ,  $R_u$  irá nomear os pacotes pertencentes à *FEC*  $\mathcal{F}$  com o *label*  $\mathcal{L}$ . Neste caso, o rótulo  $\mathcal{L}$  torna-se o *label* de saída de  $R_u$  para representar a *FEC*  $\mathcal{F}$  e  $\mathcal{L}$  torna-se o *label* de entrada de  $R_d$  para representar a *FEC*  $\mathcal{F}$ .  $\mathcal{L}$  é um valor arbitrário cuja associação à *FEC*  $\mathcal{F}$  é local ao enlace conectando  $R_u$  e  $R_d$ .

Na arquitetura MPLS, a decisão de associar um determinado *label*  $\mathcal{L}$  a uma *FEC*  $\mathcal{F}$  particular é realizada pelo *LSR* que se encontra na vante do fluxo (*downstream*). Desta maneira, os *labels* são atribuídos pelo *LSR downstream* e informados por este ao(s) *LSR(s) upstream* sobre a as-

sociação  $\mathcal{L}/\mathcal{F}$ . No caso da figura 2.4, o nó *downstream* ( $LSR_{i+1}/LSR_i$ ) aloca um rótulo ( $L_1/L_2$ ) para a *FEC*  $y$  e envia-o ao vizinho *upstream* ( $LSR_i/LSR_{i-1}$ ).

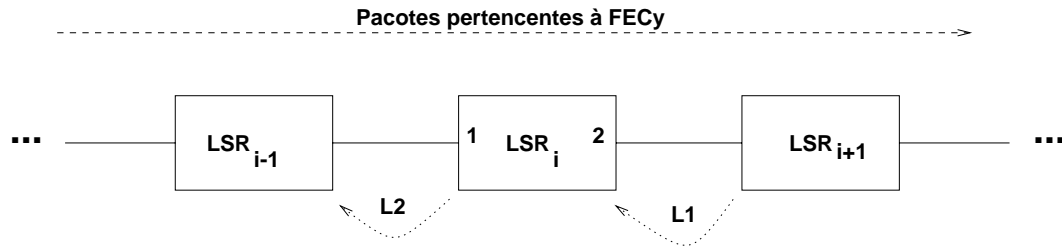


Figura 2.4: Estabelecimento de LSPs: Distribuição de Label *Downstream*

Ainda com relação à figura 2.4, na figura 2.5 encontramos um exemplo ilustrando a entrada da tabela de labels (LIB - *Label Information Base*) do  $LSR_i$  para o caso da *FEC*  $y$ .

Uma associação  $\mathcal{L}/\mathcal{F}$  (*Label/FEC*) particular distribuída de  $R_d$  para  $R_u$  pode possuir atributos. Neste caso, quando  $R_u$  for atuar como um *LSR downstream* e distribuir um *label* para a *FEC*  $\mathcal{F}$ , ele deverá também distribuir os atributos correspondentes recebidos de  $R_d$ .

Um *LSR* informa outro *LSR* sobre uma associação  $\mathcal{L}/\mathcal{F}$  através de um conjunto de procedimentos que compõem o Protocolo de Distribuição de *Labels* (LDP). Dois *LSRs* que trocam informações sobre uma associação  $\mathcal{L}/\mathcal{F}$  são denominados de pares LDP isto porque é criada uma adjacência LDP entre eles. Na realidade, o protocolo LDP compreende toda a negociação envolvendo 2 pares LDP de modo a permitir que ambos aprendam as capacidades do outro do ponto de vista do MPLS. A arquitetura MPLS admite que outros protocolos também possam participar da distribuição de *labels* como é o caso do RSVP e do BGP.

A arquitetura MPLS permite que um *LSR* solicite de forma explícita ao seu próximo *hop* a associação de um *label* para uma determinada *FEC*. Esta forma de operação é denominada de *downstream* sob demanda. A arquitetura especifica também que um *LSR* pode definir uma associação  $\mathcal{L}/\mathcal{F}$  sem que esta seja explicitamente requisitada. Esta forma de operação denomina-se *downstream*. Ambas as formas de distribuição podem ser utilizadas simultaneamente em uma mesma rede bastando somente que os *LSRs* adjacentes, do ponto de vista do protocolo LDP, definam a forma pela qual irão operar.

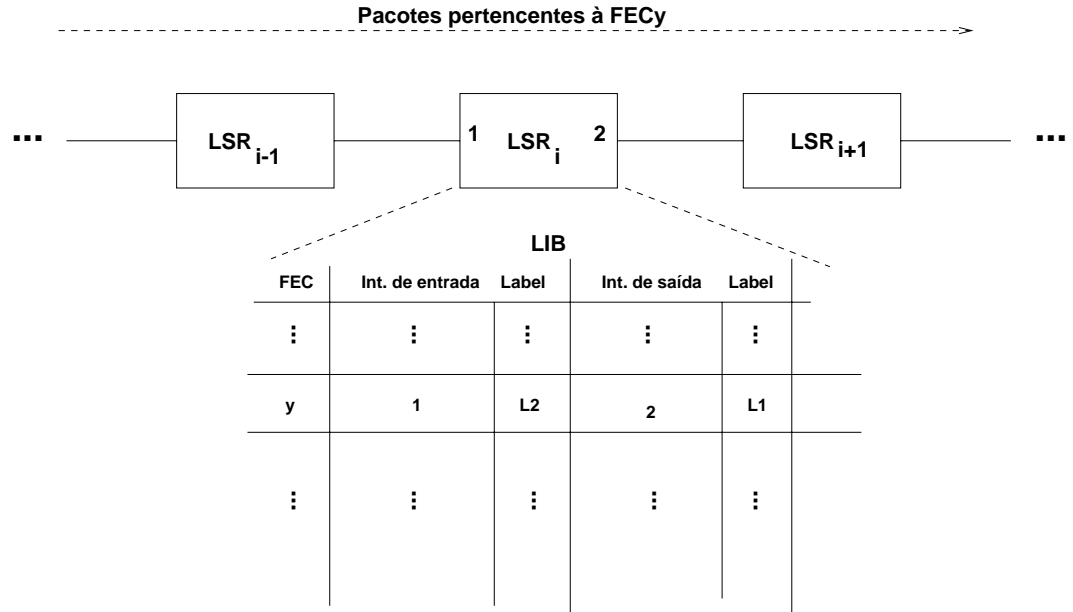


Figura 2.5: Estabelecimento de LSPs: Distribuição de Label *Downstream*

### Modo de Retenção de Label

Um  $LSR R_u$  pode receber uma associação  $\mathcal{L}/\mathcal{F}$  de um  $LSR R_d$  mesmo que este último não seja o próximo o *hop* relativamente àquela  $FEC$  do ponto de vista da tabela de roteamento do  $LSR R_u$ . Neste caso, o  $LSR R_u$  tem duas opções: 1) operar no modo de retenção de *label* liberal, no qual ele armazena a associação  $\mathcal{L}/\mathcal{F}$  recebida do  $LSR R_d$ , mesmo que este último nó não seja o próximo *hop* para a  $FEC \mathcal{F}$ ; ou 2) operar no modo de retenção de *label* conservativo, no qual a associação  $\mathcal{L}/\mathcal{F}$  recebida é descartada no caso do  $LSR R_d$  não ser o próximo *hop*. A primeira estratégia é vantajosa no caso de alterações no roteamento, isto porque se o  $LSR R_d$  tornar-se o próximo *hop* para a  $FEC \mathcal{F}$ , o  $LSR R_u$  poderá encaminhar os pacotes imediatamente utilizando o *label*  $\mathcal{L}$  armazenado; por outro lado, a vantagem da solução 2 consiste em utilizar um espaço de *labels* mais reduzido.

### A pilha de labels

A arquitetura MPLS permite que um pacote carregue vários *labels* organizados na forma de uma pilha de *labels* onde o último *label* associado é o

primeiro *label* a ser eliminado. A questão fundamental envolvendo a pilha de *labels* é que a decisão de encaminhamento realizada nos *LSRs* é sempre baseada no *label* que se encontra no topo da pilha. Um dos aspectos fundamentais da pilha de *labels* consiste na possibilidade de definir uma hierarquia de *labels*, hierarquia que possui implicações importantes do ponto de vista do roteamento e do tunelamento de informações. Entretanto, o processamento do *label* é sempre baseado naquele que se encontra no topo da pilha, sem levar em conta a possibilidade de que algum outro *label* possa ter estado no topo no passado, ou mesmo, que outros *labels* encontrem-se, no presente, abaixo na pilha.

Um pacote sem *labels* pode ser considerado como um pacote no qual a pilha de *labels* possui profundidade igual a zero. Caso a pilha de *labels* possua profundidade  $m$ , define-se que o *label* no topo da pilha corresponde ao nível  $m$  e o *label* que se encontra no final da pilha possui o nível 1.

### Entrada na Tabela de Encaminhamento

O NHLFE (*Next Hop Label Forwarding Entry*) corresponde à entrada na Tabela de Encaminhamento e é utilizado no momento do encaminhamento do pacote rotulado. O conteúdo desta entrada possui as seguintes informações:

1. O próximo *hop*;
2. A operação a ser realizada na pilha de *labels* contida no pacote. As operações possíveis são:
  - substituir o *label* que se encontra no topo da pilha por um novo *label*;
  - retirar o *label* que se encontra no topo da pilha (*pop*);
  - substituir o *label* que se encontra no topo da pilha por um novo *label* e, em seguida, empurrar (*push*) um (ou mais) novo(s) *label*(s) na pilha.

A entrada na tabela pode conter ainda as seguintes informações:

- o encapsulamento no enlace de dados a ser utilizado quando da transmissão do pacote;
- a forma de codificação da pilha de *labels* quando da transmissão do pacote;

- qualquer outra informação necessária para o encaminhamento adequado do pacote.

É possível que a entrada NHLFE indique que o próximo *hop* é o próprio *LSR* que está examinando o pacote para fins de encaminhamento. Neste caso, é realizada uma operação de *pop*, ou seja, é retirado o *label* que se encontra no topo da pilha. Após esta operação, o *LSR* deve realizar uma nova decisão de encaminhamento baseado no que restou no pacote. É possível que ainda seja um pacote rotulado, ou então um pacote na forma nativa. Neste último caso, o *LSR* deve operar sobre o cabeçalho do pacote IP para realizar o seu encaminhamento.

Algumas operações envolvendo a entrada NHLFE são definidas a seguir:

- ILM (*Incoming Label Map*): é o mapeamento dos *labels* de entrada nas respectivas NHLFEs. É utilizado quando do recebimento de pacotes rotulados;
- FTN (*FEC-to-NHLFE Map*): é o mapeamento de *FECs* em NHLFEs. Esta operação é utilizada quando do encaminhamento de pacotes que são recebidos sem rótulos mas que devem ser rotulados antes de serem encaminhados;
- Comutação de *Label* (*Label Swapping*): o *LSR* analisa o *label* que se encontra no topo da pilha. Ele utiliza a função ILM para mapear o *label* na NHLFE correspondente. A partir das informações na NHLFE o *LSR* determina a interface de saída onde o pacote deverá ser encaminhado e realiza a operação definida no NHLFE com relação à pilha de *labels*. O novo *label* é codificado na pilha contida no pacote e, em seguida, o pacote é encaminhado.

No caso do recebimento de um pacote não-rotulado, o *LSR* analisa o cabeçalho do nível de rede do pacote para determinar a *FEC* à qual ele pertence. Em seguida, o *LSR* evoca a função FTN que mapeia a *FEC* na NHLFE correspondente. Através das informações da NHLFE o *LSR* decide onde encaminhar o pacote e realiza a operação especificada no topo da pilha de *label* com exceção, neste caso, da operação de *pop* que seria ilegal para um pacote não-rotulado.

Uma observação que deve ser destacada na comutação por *label* é que o próximo *hop* é sempre aquele indicado pela NHLFE correspondente à *FEC* do pacote. Este *hop* pode ser diferente do próximo *hop* indicado pelo roteamento IP. Um exemplo desta situação pode ocorrer quando da utilização do roteamento explícito.



**Caminhos Comutados por Labels (LSP-Label Switch Path)**

Um *LSP* de nível  $m$  relativamente a um determinado pacote  $P$  corresponde a uma sequência de roteadores  $\langle R_1, \dots, R_n \rangle$  que apresentam as seguintes propriedades:

1.  $R_1$  é o *LSR* de ingresso do *LSP* (*LSP ingress*) e responsável pela colocação (*push*) de um *label* na pilha de *labels* do pacote  $P$  resultando em uma pilha de profundidade  $m$ ;
2. para qualquer  $i$ , onde  $1 < i < n$ , o pacote  $P$  possui uma pilha de *labels* de profundidade  $m$  quando recebido pelo *LSR* $_i$ . Em nenhum instante, durante o caminho de  $P$  entre  $R_1$  e  $R_{n-1}$ , a sua pilha de *labels* possui uma profundidade menor do que  $m$ ;
3. para qualquer  $i$ ,  $1 < i < n$ ,  $R_i$  transmite o pacote  $P$  para  $R_{i+1}$  através da técnica MPLS, ou seja, através do *label* que se encontra no topo da pilha de *labels* o qual serve como um índice para a função *ILM*;
4. a saída do *LSP* corresponde ao *LSR* que realiza uma decisão de encaminhamento baseada na comutação de um *label* de nível  $m - k$ , onde  $k > 0$ , ou quando a decisão de encaminhamento é realizada por outros procedimentos de encaminhamento, ou seja, não são procedimentos MPLS.

As especificações anteriores indicam que sempre que um *LSR* coloca um *label* em um pacote já rotulado, é preciso que o novo *label* corresponda a uma *FEC*  $\mathcal{F}$  cuja saída do *LSP* corresponde a um *LSR* que atribuiu o *label* que agora é o segundo na pilha.

Consideremos o conjunto de nós que podem ser nós de ingresso de um *LSP* para a *FEC*  $\mathcal{F}$ . Desta maneira, existirá um *LSP* para a *FEC*  $\mathcal{F}$  começando em cada um destes nós. No caso de um certo número destes *LSPs* possuir o mesmo *LSR* de saída do *LSP*, podemos considerar este conjunto de *LSPs* como uma árvore que possui o *LSR* de egresso como raiz, caracterizando aquilo que se denomina de uma árvore multiponto-ponto, ou seja, uma árvore *LSP* (*LSP tree*) para a *FEC*  $\mathcal{F}$ .

**Pop pelo Penúltimo Hop**

Supondo que  $\langle R_1, \dots, R_n \rangle$  é um *LSP* de nível  $m$  para o pacote  $P$ , este pode ser transmitido do *LSR*  $R_{n-1}$  para o *LSR*  $R_n$  (*LSP* de egresso) com um *label* de profundidade  $m - 1$ . Isto indica que a pilha de *labels* pode ter

o *label* do topo retirado (*pop*) pelo penúltimo *LSR* do *LSP* ao invés de ser retirado pelo *LSR* de egresso. A partir do instante que  $R_{n-1}$  decide enviar o pacote para  $R_n$ , o *label* que está no topo da pilha de *labels* não tem mais função, não precisando ser transportado no pacote.

Esta ação tem um sentido prático importante porque, em caso contrário, o *LSP* de egresso deve realizar 2 buscas: na primeira, ao examinar o *label* no topo da pilha ele será indicado como o próximo *hop* levando-o a realizar um *pop* na pilha; caso o pacote continue rotulado, o *LSR* de egresso deverá fazer uma nova busca na tabela para determinar o próximo *hop*; caso o pacote não continue rotulado, o *LSR* de egresso deverá fazer um encaminhamento baseado no cabeçalho do pacote. Quando o penúltimo *hop* realiza o *pop* na pilha de *labels*, o *LSR* de egresso realiza uma única busca para encaminhamento do pacote, esteja este rotulado ou não.

Uma consequência do *pop* feito pelo penúltimo *hop* é que o *LSP* de egresso não precisa ser um *LSR*. A operação de *pop* pelo penúltimo *hop* deve ser explicitamente solicitada pelo nó de egresso ou, ainda, no caso do próximo *hop* no *LSP* não suportar MPLS. As negociações iniciais do *LSP* permitem os *LSRs* determinarem se o penúltimo *hop* fará ou não a retirada do *label*.

### **Próximo Hop LSP**

O próximo *hop LSP*, para um determinado pacote rotulado, é o *LSR* indicado como próximo *hop* pela entrada NHLFE usada para o encaminhamento do pacote. É importante observar, como já comentado anteriormente, que o próximo *hop* pode ser diferente do próximo *hop* correspondente ao roteamento nível 3, este último denominado de próximo *hop* L3 (NHL3).

De modo a evitar comportamentos indesejáveis na rede como, por exemplo, a ocorrência de *loops*, quando um pacote rotulado é recebido em um *LSR* contendo no topo da pilha um *label* considerado inválido o pacote deve ser descartado.

### **Controle de LSP: ordenado x independente**

O tipo de *FEC* mais comumente referenciada no MPLS corresponde a prefixos de endereços que são distribuídos através de algoritmos de roteamento. Para este tipo de *FEC*, a definição do *LSP* pode-se dar de duas formas: independente e ordenado.

No modo de controle independente cada *LSR*, ao detectar uma *FEC*  $\mathcal{F}$ , toma uma decisão independente de alocar um *label*  $\mathcal{L}$  àquela *FEC*  $\mathcal{F}$  e de

distribuir esta associação  $\mathcal{L}/\mathcal{F}$  aos seus pares *LDP upstream*.

No caso do controle ordenado, um *LSR* somente associa um *label* a uma *FEC* caso ele seja o *LSR* de egresso para aquela *FEC*, ou caso ele já tenha recebido uma associação de *label* do próximo *hop* correspondente à *FEC*.

O controle ordenado é necessário quando é preciso que uma determinada *FEC* possua um conjunto específico de propriedades segundo um determinado caminho.

No caso do controle independente é possível que alguns *LSRs* comecem a comutar o tráfego de uma determinada *FEC* antes do *LSP* ser completamente estabelecido, neste caso o tráfego pode seguir um caminho que não possui as propriedades desejadas.

### Agregação

A forma mais imediata de particionamento do tráfego em *FECs* consiste em definir uma *FEC* para cada prefixo de endereço que aparece na tabela de roteamento. Esta abordagem pode gerar, entretanto, um conjunto de *FECs* para as quais os respectivos tráfegos percorrem a mesma rota. Por exemplo, um conjunto de prefixos diferentes pode ter o mesmo roteador de saída, neste caso, a comutação de *label* é utilizada para encaminhar os pacotes até este nó de saída. Em uma situação como esta, é interessante criar uma nova *FEC* que corresponda à união das *FECs* anteriores e utilizar um único *label* para representar esta nova *FEC* e, conseqüentemente, a união das várias *FECs*. Esta ação corresponde ao conceito de agregação, o que permite reduzir o número de *labels* e, conseqüentemente, o tráfego de controle representado pelo LDP.

Dado um conjunto de *FECs* passíveis de agregação em uma única *FEC*, são possíveis os seguintes níveis de agregação:

1. agregar o conjunto de *FECs* em uma única *FEC* (maior nível de agregação);
2. agregar o conjunto de *FECs* em um outro conjunto de *FECs*, este último provavelmente contendo um número menor de *FECs* do que o conjunto anterior;
3. não agregar o conjunto de *FECs* (menor nível de agregação).

Quando é utilizado o controle ordenado no estabelecimento do *LSP*, cada *LSR* deve adotar, para um determinado conjunto de *FECs*, a granulação usada pelo seu próximo *hop* relativamente às aquelas *FECs*. Quando é

utilizado o controle independente, é possível que *LSRs* adjacentes agreguem conjuntos de *FECs* de formas diferentes.

### **Seleção de Rota**

A seleção de rota corresponde ao método utilizado para seleção do *LSP* relativamente a uma determinada *FEC*. A arquitetura MPLS suporta 2 opções para seleção de rotas:

1. roteamento *hop-by-hop*;
2. roteamento explícito.

No roteamento *hop-by-hop*, cada nó escolhe de forma independente o próximo *hop* para cada *FEC*, o que corresponde à forma tradicional do roteamento IP. Um *LSP* roteado *hop-by-hop* é um *LSP* cuja rota é selecionada pelo roteamento *hop-by-hop*.

No caso do *LSP* roteado explicitamente, cada *LSR* não escolhe o próximo *hop* de forma independente. Na realidade, um único *LSR*, normalmente o *LSR* de ingresso ou o *LSR* de egresso, especifica todas (ou parte) dos *LSRs* que fazem parte do *LSP*. Caso sejam especificados todos os *LSRs* que constituem o *LSP*, diz-se que se trata de um *LSP* roteado explicitamente de forma estrita ou, no caso de serem especificados somente alguns dos *LSRs* que fazem parte do *LSP*, diz-se que se trata de um *LSP* explicitamente roteado fracamente.

Normalmente, o roteamento explícito pode ser realizado por configuração ou dinamicamente, neste último caso baseado em protocolos de roteamento que operam com o conceito do estado dos enlaces.

O maior interesse representado pelo roteamento explícito consiste na possibilidade de realização de engenharia de tráfego. Como no MPLS o roteamento explícito é criado no momento da distribuição de *labels*, não há necessidade de explicitar a rota em cada pacote IP como ocorre atualmente através do campo de opções do cabeçalho IP. Desta forma, o roteamento explícito do MPLS é muito mais eficiente do que o correspondente no caso do IP representado pelo roteamento na origem (*source routing*).

### *TTL - Time-to-Live*

O protocolo IP trata o campo TTL no cabeçalho do pacote através do decremento do seu conteúdo do valor 1 todas as vezes que o pacote passa por um roteador. Caso o valor do TTL torne-se 0 antes de alcançar o

seu destino, o pacote é descartado. A função do campo TTL consiste no controle de *loops* e no suporte ao comando *traceroute*. Este último permite determinar quantos *hops* os pacotes devem percorrer entre 2 pontos na rede. No caso do MPLS, o pacote ao atravessar um *LSP* deve possuir na saída do *LSP* um valor de TTL idêntico ao que teria caso tivesse atravessado a mesma sequência de roteadores sem sofrer a comutação baseada em *labels*. A forma como o valor de TTL é tratado no MPLS depende da maneira pela qual o *label* é transportado, podendo assumir 2 formas:

1. o *label* é transportado em um cabeçalho MPLS (*Shim Header*) inserido entre os cabeçalhos de enlace e de rede (figura 2.6). O encapsulamento possui os seguintes campos:
  - um rótulo de 20 bits;
  - um campo de Classe de Serviço (CoS) formado por 3 bits;
  - um campo TTL de 8 bits;
  - um campo B de 1 bit cujo objetivo é indicar se o rótulo corresponde, ou não, ao último de uma pilha de rótulos, permitindo o encapsulamento de múltiplos rótulos.
2. o *label* é transportado em um cabeçalho de nível 2 como nos casos do ATM e do Frame-Relay. Para o ATM os pacotes são encapsulados na AAL5 e segmentados em células. O rótulo estará presente em cada célula resultante da segmentação (figura 2.7). Rótulos para os *LSRs* Frame-Relay são indicados no DLCI.

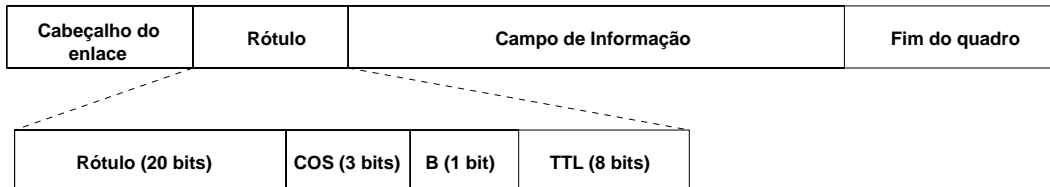


Figura 2.6: Encapsulamento genérico: posicionado entre os cabeçalhos dos níveis 2 e 3

No primeiro caso, onde o *label* encontra-se entre os cabeçalhos de nível 2 e nível 3, MPLS deve oferecer um campo TTL que será inicialmente carregado com o valor do campo TTL do cabeçalho de rede e será decrementado em

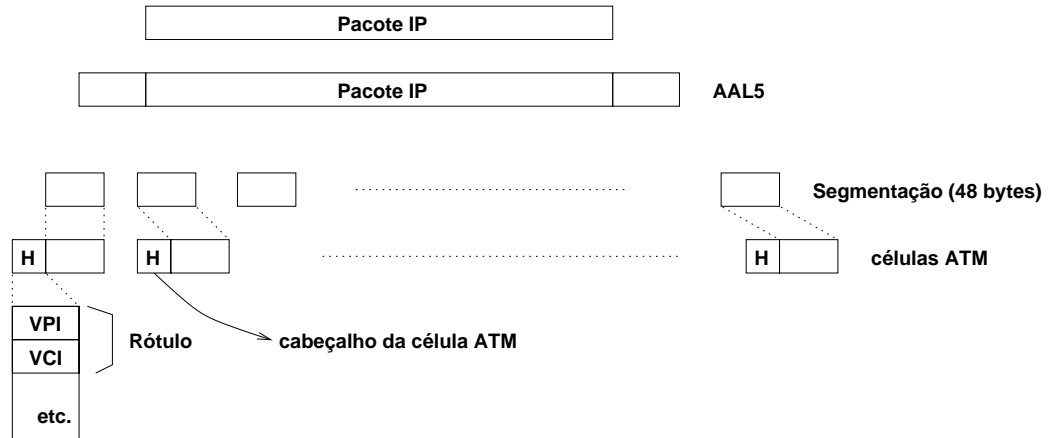


Figura 2.7: Rótulo para *LSRs*-ATM: campos VPI & VCI

cada *hop LSR*. Quando o pacote emergir do *LSP* o valor do campo TTL inserido entre os cabeçalhos de enlace e de rede deverá ser copiado para o campo do TTL do cabeçalho de nível 3.

No segundo caso, típico das tecnologias ATM e Frame-Relay, não é possível decrementar o campo TTL em cada *hop LSR* porque o encapsulamento de nível 2 nestes casos não possui um campo TTL para ser decrementado. Um segmento *LSP* formado de uma sequência de *LSRs* não capazes de decrementar o TTL do pacote denomina-se de segmento *LSP* não-TTL (*non-TTL LSP segment*). Entretanto, também neste caso é necessário que o TTL do pacote ao emergir de um segmento *LSP* não-TTL reflita o número de *hops LSR* atravessados. No caso *unicast*, isto pode ser obtido pela propagação aos nós de ingresso de uma informação contendo o comprimento do *LSP*. Isto pode ser obtido através de procedimentos que suportam a funcionalidade do comando *traceroute*, por exemplo, através do encaminhamento *hop-by-hop* dos pacotes. De posse deste valor, quando do ingresso do pacote em um segmento *LSP* não-TTL, o valor do TTL do pacote deverá ser decrementado deste comprimento. Caso no ingresso seja detectado que o TTL do pacote irá expirar antes de alcançar a saída do segmento não-TTL, o pacote não deverá ser comutado por *label*.

### Codificação de *Labels*

Com o objetivo de permitir a transmissão da pilha de *labels* pelo pacote, é necessária a definição de uma codificação para a pilha. Várias formas de codificação são possíveis e a escolha de qual utilizar depende do hardware que está sendo empregado para o encaminhamento dos pacotes:

- Hardware/Software específicos para MPLS: neste caso a melhor solução é definir um protocolo novo e utilizá-lo como uma *cunha* (*shim*) entre os cabeçalhos dos níveis de enlace e rede. Na realidade, esta operação corresponde a um novo encapsulamento para o pacote da camada de rede. O encapsulamento deve ser genérico e independente de protocolo de modo a permitir a sua utilização no encapsulamento de qualquer camada de rede. Este encapsulamento genérico será, por sua vez, encapsulado em protocolo de camada de enlace (figura 2.6);
- Utilização de comutadores ATM como *LSRs*: podemos observar que o encaminhamento de pacotes conforme proposto pela tecnologia MPLS, assemelha-se à forma de comutação de tecnologias anteriores como o ATM e o Frame-Relay. No caso do ATM, a identificação da porta de entrada e os valores de VPI/VCI atuam como um índice em uma tabela de comutação que permite obter a interface de saída e o VPI/VCI (ou seja, o novo *label*!) de saída. Desta maneira, se um ou mais *labels* puderem ser codificados nos campos destes comutadores, eles poderão atuar como *LSRs* através de uma adaptação do seu software de controle. Neste caso estes comutadores são denominados de *LSRs-ATM*.

Supondo, no caso do ATM, a utilização da camada AAL5, existem 3 formas possíveis de codificação de *labels*:

1. Codificação SVC: utiliza o VPI/VCI para codificação do *label* que se encontra no topo da pilha de *labels*. Neste caso, o *LSP* corresponde a um SVC (*Switched Virtual Channel*) e o protocolo LDP atua como a sinalização ATM. Deve ser destacado que nesta codificação não é possível realizar as operações de *push* e *pop* relativamente à pilha de *labels*;
2. Codificação SVP (*Switched Virtual Path*): o campo VPI é utilizado para codificar o *label* que se encontra no topo da pilha, e o campo VCI é utilizado para o segundo *label* da pilha caso este esteja presente. Este

tipo de codificação permite, diferentemente da codificação anterior, o uso da comutação VP (*Virtual Path*). Neste caso, o *LSP* corresponde a um SVP ATM, e o LDP serve como um protocolo de sinalização para o ATM. A codificação permite ainda que o *LSR-ATM* de egresso do VP realize a operação de *pop*.

3. Codificação SVC multiponto: o campo VPI é empregado para codificar o *label* que se encontra no topo da pilha e parte do VCI é empregada para codificar, caso este esteja presente, o segundo *label* da pilha. O restante do campo VCI é empregado para identificar o LSR de ingresso daquele LSP. Nestas condições é possível implementar VPs multiponto-ponto, isto porque é possível demultiplexar as células de origens diferentes em um mesmo VP em função do campo que permite identificar o LSR de ingresso.

Esta técnica depende de capacidade em atribuir valores de VCI de 16 bits para cada comutador ATM de modo que um único valor de VCI não seja atribuído a 2 comutadores diferentes.

#### **Agregação de Labels (*Label Merging*)**

Um LSR é capaz de realizar a agregação de *labels* se ele pode receber 2 (ou mais) pacotes de interfaces de entrada diferentes, com *labels* diferentes ou não, e enviá-los na mesma interface de saída com o mesmo *label* gerando um *LSP Multiponto – ponto*. Após a transmissão destes pacotes a informação de que eles foram recebidos a partir de interfaces diferentes, com *labels* diferentes ou não, é perdida (figura 2.8).

No caso do ATM, a agregação de *labels* resulta na intercalação de células pertencentes a pacotes diferentes inviabilizando a remontagem dos pacotes no destino. Existem 2 métodos que podem ser utilizados para eliminar a intercalação de células no ATM e permitir que comutadores ATM realizem a agregação de fluxo:

1. Merge de VPs: utiliza a codificação Multiponto-ponto permitindo que vários caminhos virtuais sejam agregados em um caminho virtual único. As fontes dos pacotes podem ser diferenciadas e, conseqüentemente, permitir a recuperação dos pacotes originais através do uso de VCIs diferentes dentro do VP (figura 2.9);
2. Merge de VC: neste caso, os comutadores devem armazenar as células de um pacote até que o pacote inteiro seja recebido, o que pode ser detectado pelo indicador de fim de quadro da AAL5 (figura 2.10).



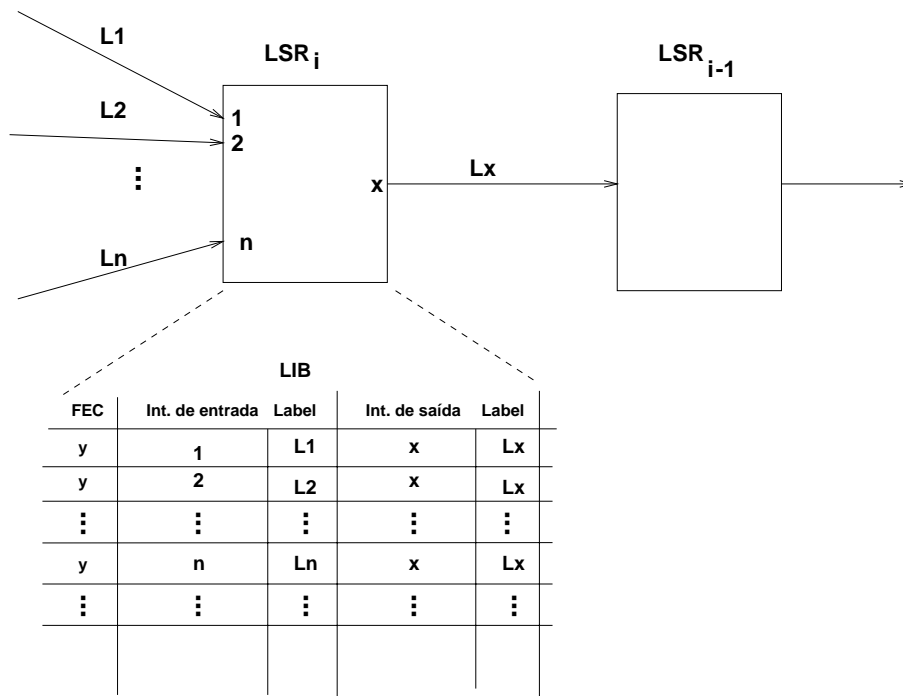


Figura 2.8: Merge de Labels: LSP Multiponto-ponto

O método baseado no merge de VP possui a vantagem de ser compatível com um grande número de comutadores ATM, além de não possuir os atrasos nos pontos de agregação e requisitos de armazenamento como no caso do merge de VCs. As desvantagens no caso do merge de VPs consiste no espaço limitado do campo VPI e na necessidade de coordenar o espaço de VCI para cada VP.

Cada comutador ATM participando de uma infra-estrutura MPLS deve conhecer se os seus vizinhos ATM imediatos realizam o merge de VPs, o merge de VCs ou nenhum tipo de merge.

### Túneis e Hierarquia

Algumas situações demandam que um roteador  $R_u$  atue de modo a fazer com que um pacote seja encaminhado a um roteador  $R_d$ , mesmo que  $R_u$  e  $R_d$  não sejam roteadores consecutivos. Isto pode ser realizado no IP através do encapsulamento do pacote em um pacote cujo endereço de destino é

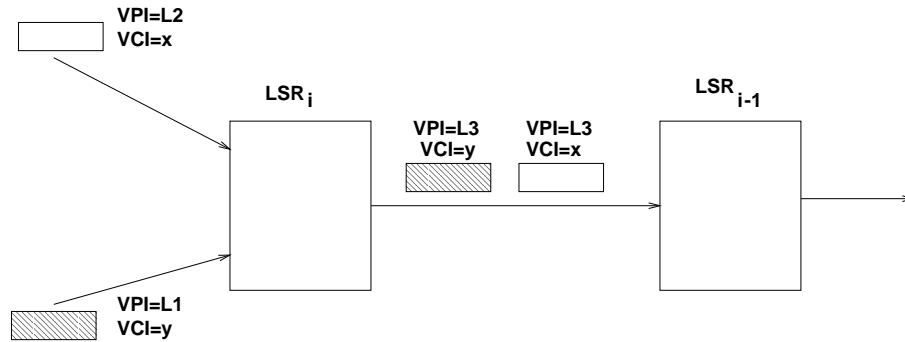


Figura 2.9: Merge de VPs (VP-Merge)

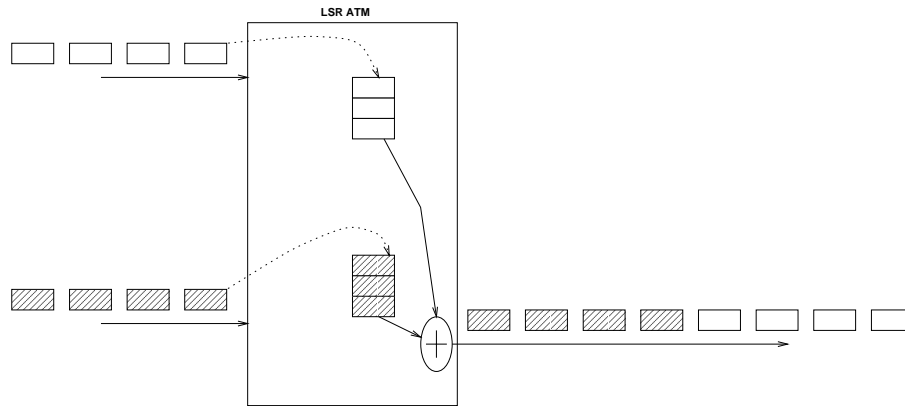


Figura 2.10: Merge de VCs (VC-Merge)

$R_d$ , criando o que é denominado de túnel entre  $R_u$  e  $R_d$ . Caso o pacote tunelado seja o caminho *hop-by-hop* de  $R_u$  a  $R_d$ , diz-se que se trata de um túnel roteado *hop-by-hop* cujo ponto\_extremo de transmissão é  $R_u$  e o ponto\_extremo de recepção é  $R_d$ .

No caso do MPLS é possível implementar o tunelamento através do *LSP* e utilizar a comutação por *label*, ao invés do encaminhamento de pacote no nível de rede, para fazer com que o pacote atravessasse o túnel. O túnel deve ser formado por um *LSP*  $\langle R_1, \dots, R_n \rangle$ , onde  $R_1$  é o ponto\_extremo de transmissão do túnel e o  $R_n$  é o ponto\_extremo da recepção do túnel. Neste caso trata-se de um túnel *LSP*.

O critério para atribuir um determinado pacote a um túnel *LSP* é uma

decisão local no ponto\_extremo de transmissão do túnel. Para enviar o pacote em um túnel *LSP*, o LSR de entrada no túnel introduz (*push*) um *label* para o túnel na pilha de *labels* e envia o pacote rotulado para o próximo *hop* no túnel. Conforme já mencionado em outra parte deste texto, a pilha de *label* pode ter o *label* do topo retirado (*popped*) no penúltimo LSR do túnel.

O conceito de hierarquia no MPLS pode ser visto como túneis *LSPs* dentro de *LSPs*. Consideremos, por exemplo, um  $LSP < R_1, R_2, R_3, R_4, R_5 >$  e que  $R_1$  recebeu um pacote  $P$  não rotulado (figura 2.11).  $R_1$  coloca um *label* na pilha de *labels* do pacote e encaminha-o para  $R_2$ . Suponhamos ainda que  $R_2$  e  $R_3$  não estão diretamente conectados, mas são vizinhos pelo fato de serem os pontos extremos de um túnel *LSP*, e que a sequência real de LSRs atravessados por  $P$  é  $< R_1, R_2, R_{21}, R_{22}, R_{23}, R_3, R_4, R_5 >$ . Quando  $P$  viaja de  $R_1$  a  $R_2$ , ele possui uma pilha de *label* de profundidade 1.  $R_2$  ao realizar a comutação do *label* substitui o *label* de entrada com o *label* de saída que possui significado para  $R_3$  e detecta que *label* deve entrar no túnel. Consequentemente, um novo *label* é introduzido (*push*) na pilha de *labels*. Este *label* de nível 2 possui um valor que tem significado para  $R_{21}$ . A partir deste ponto a comutação realiza-se no *label* de nível 2 através de  $R_{21}$ ,  $R_{22}$  e  $R_{23}$ . Este último, que corresponde ao penúltimo *hop* no túnel  $R_2 - R_3$ , retira (*pop*) o *label* da pilha antes de encaminhar o pacote para  $R_3$ . Quando  $R_3$  recebe o pacote este contém um *label* de nível 1, indicando que o pacote saiu do túnel.  $R_3$  encaminha o pacote rotulado de nível 1 a  $R_4$  e como este é o penúltimo *hop* com relação ao *LSP* de nível 1 do pacote, ele retira (*pop*) o *label* e encaminha o pacote não rotulado para  $R_5$ .

O mecanismo baseado na pilha de *labels* permite criar tunelamento *LSP* com vários níveis de profundidade. No caso do exemplo anterior, do ponto de vista do *LSP* de nível 2, o par LDP de  $R_2$  é  $R_{21}$ . Do ponto de vista do *LSP* de nível 1, os pares LDP de  $R_2$  são  $R_1$  e  $R_3$ . Neste caso,  $R_2$  e  $R_{21}$  devem ser vizinhos do ponto de vista do protocolo IGP (*Internet Gateway Protocol*) enquanto  $R_2$  e  $R_3$  não possuem esta relação. Quando dois *LSRs* são vizinhos IGP, eles são referenciados como par LDP local. Quando 2 *LSRs* podem ser um par LDP mas não são vizinhos IGP, eles são denominados de par LDP remoto. No exemplo anterior  $R_2$  e  $R_{21}$  são um par LDP local e  $R_2$  e  $R_3$  são um par LDP remoto.

### 2.3.1 Mecanismos de Alocação de Rótulos

Para fazer a distribuição de rótulos através dos roteadores MPLS é necessário empregar um protocolo específico. Dentro do grupo de trabalho de

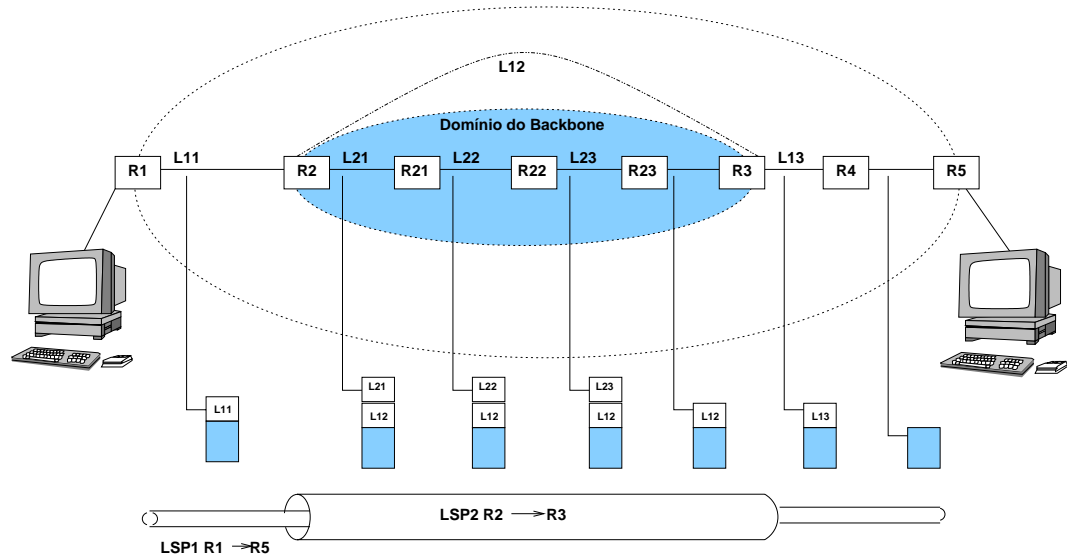


Figura 2.11: Empilhamento de Rótulos: Túneis e Hierarquias

MPLS definiu-se um novo protocolo especialmente para servir ao propósito de alocação de rótulos em redes MPLS. Este é denominado *Label Distribution Protocol - LDP*. Há também, dentro deste mesmo grupo de trabalho, discussões envolvendo o uso do protocolo RSVP como mecanismo de distribuição de rótulos. As propostas que existem neste sentido é que RSVP e MPLS sejam combinados com roteamento explícito, e que sejam usados para engenharia de tráfego.

A seguir apresentam-se os mecanismos básicos de distribuição de rótulos via LDP.

### Label Distribution Protocol - LDP

Um conceito fundamental no MPLS é que 2 *LSRs* devem concordar no significado dos *labels* utilizados para encaminhar o tráfego entre eles e através deles. Este entendimento comum é obtido através do protocolo LDP. LDP é o protocolo que efetivamente cria os LSP's através do mapeamento de informações da camada de rede (por exemplo, o roteamento) diretamente em caminhos comutados na camada de enlace de dados. À medida que o LDP distribui rótulos para uma dada FEC, cria-se uma conexão fim-a-fim para o envio de pacotes. A *FEC* associada ao LSP especifica quais pacotes

são mapeados àquele LSP. Dois *LSRs* que se comunicam para a troca de informações de rótulos são denominados "pares LDP" (*LDP Peers*), e diz-se haver uma sessão LDP entre eles.

Se dois pares LDP se conectam via uma interface ATM, uma conexão não MPLS deverá estar sempre disponível, capaz de transportar pacotes IP não rotulados. Esta conexão não MPLS deverá ser empregada para transportar pacotes LDP entre os dois pares LDP, e pode também ser usada para transportar qualquer outro pacote não rotulado (tais como pacotes do protocolo OSPF, etc.). O encapsulamento LLC/SNAP do RFC 1483 deverá ser usado na conexão não MPLS.

Há quatro categorias de mensagens LDP:

1. Mensagens de descoberta: são empregadas inicialmente para anunciar a presença de um LSR em uma rede e posteriormente para ratificar que o LSR continua presente;
2. Mensagens de Sessão: são usadas para estabelecer, manter e terminar sessões LDP entre dois LSR's;
3. Mensagens de Anúncio de Rótulo: são usadas para criar, modificar e suprimir mapeamentos de rótulos para FEC's;
4. Mensagens de Notificação: usadas para prover informações de estado da rede e sinalizar erros.

O estabelecimento de uma sessão LDP se dá nas seguintes fases:

- Descoberta do Par LDP: Antes que dois LSR's adjacentes possam iniciar a troca de mensagens para a distribuição de rótulos, eles devem estabelecer uma relação de vizinhança. Para descobrir que um roteador vizinho suporta o protocolo LDP, um *LSR* envia mensagens *HELLO*. Estas mensagens são enviadas como pacotes UDP, normalmente para o endereço de grupo multicast "todos roteadores LSR". No entanto, há casos onde se deseja estabelecer uma sessão LDP com um LSR que não é um vizinho local. Isto ocorre, por exemplo, quando dois roteadores BGP localizados na borda de um domínio, que não têm conexão direta entre si, desejam trocar informações de rótulos. Neste caso, a mensagem *HELLO* é denominada *targeted HELLO*, e é enviada com o endereço IP do LSR com o qual se deseja estabelecer uma relação de par LDP (pares LDP remotos). Esta questão de pares LDP remotos é importante, como vimos anteriormente, na criação de túneis;

- Conexão da Camada de Transporte: Uma vez descoberta a presença de outro LSR suportando o LDP, dois LSR's pares iniciam o estabelecimento de uma sessão de comunicação TCP. Para o protocolo LDP operar corretamente, as informações do LDP precisam ser transmitidas de forma confiável e as mensagens LDP pertencentes a uma dada FEC precisam ser transmitidas em sequência. Também é necessário controle de fluxo, bem como transportar múltiplas mensagens LDP em um único datagrama. Para atender a estes requisitos, define-se o TCP como mecanismo de transporte para o LDP;
- Inicialização da Sessão LDP: Após o estabelecimento de uma sessão TCP para o transporte do protocolo LDP, o lado que estiver controlando a inicialização da sessão enviará uma mensagem *INITIALIZATION*. Esta mensagem contém diversas informações quanto às características suportadas pelo LSR: versão do protocolo LDP, método de distribuição de rótulo (*downstream*, *downstream on demand*), valores de *timers*, faixa de valores de VPI/VCI a ser empregada no caso de LSR's ATM, tempo de *hold* para as mensagens de *KEEP\_ALIVE*, se suporta ou não detecção de *loop*, se suporta *merge* e qual tipo (VP merge, VC merge, ou ambos), etc. O outro lado pode ou não aceitar os valores propostos na mensagem *INITIALIZATION*:
  - Caso discorde, deverá enviar uma mensagem *NOTIFICATION* com indicação de *NAK* em resposta à mensagem *INITIALIZATION* recebida e encerrar a sessão;
  - Caso concorde, poderá enviar em resposta uma mensagem *INITIALIZATION* com seus parâmetros próprios (não necessariamente idênticos aos parâmetros recebidos do outro lado), seguida de uma mensagem *KEEP\_ALIVE* sinalizando aceitação dos parâmetros propostos.

O lado que controla a inicialização, se receber uma mensagem *INITIALIZATION* como resposta, deverá verificar se os parâmetros próprios propostos pelo outro lado podem ser aceitos. Caso afirmativo, ele responderá com uma mensagem *KEEP\_ALIVE* e a sessão LDP estará operacional. Caso não aceite os parâmetros propostos, enviará uma mensagem *NAK (NOTIFICATION)* e encerrará a sessão.

A mensagem de *KEEP\_ALIVE* também é enviada quando os dois pares LDP não têm mensagens para serem trocadas mas desejam manter a sessão LDP ativa. Neste caso, ela é enviada

periodicamente a um intervalo definido pelos parâmetros presentes na mensagem *INITIALIZATION*.

- **Fase Ativa:** Uma vez estabelecida a sessão LDP, os dois *LSR's* encontram-se na fase ativa para a distribuição e solicitação de criação de rótulos. Conforme discutido anteriormente, os métodos de atribuição de módulos correspondem quanto ao modo de alocação de rótulo (*downstream* e *downstream* sob demanda) e ao modo de controle de distribuição (modo de controle independente e modo de controle ordenado).

### Mensagens para Criação e Supressão de Rótulos

Uma vez estabelecida a sessão LDP entre dois *LSR's* inicia-se a fase ativa de criação e supressão de rótulos. Para a criação de um rótulo (mapeamento de um rótulo a uma *FEC*) no método de alocação *downstream*, um *LSR downstream* deve enviar uma mensagem *LABEL\_MAPPING* ao *LSR upstream*. A mensagem *LABEL\_MAPPING* contém informações do tipo:

- *FEC*;
- rótulo atribuído;
- classe de Serviço associada ao LSP em questão;
- contador de número de hops neste LSP;
- *path vector*.

Mais tarde será visto que *path vector* será empregado no algoritmo de detecção de *loops*.

Como o LDP emprega o TCP como protocolo de transporte, dispensa-se o envio de uma mensagem de reconhecimento em resposta à mensagem de *LABEL\_MAPPING*.

No caso do método de alocação *downstream sob demanda*, o nó *upstream* deve enviar uma mensagem *LABEL\_REQUEST* ao nó *downstream* contendo as seguintes informações:

- *FEC* para a qual se deseja a associação de um rótulo;
- Classe de Serviço para o LSP que se está criando;
- contador de número de hops neste LSP.

O nó *downstream* enviará em resposta uma mensagem *LABEL\_MAPPING*. Dependendo do modo de controle de distribuição (Independente ou Ordenado), a mensagem de *LABEL\_MAPPING* poderá ser enviada imediatamente (modo Independente), ou terá que aguardar até que receba do nó *downstream* a associação de um rótulo para aquela FEC (modo ordenado). Em qualquer dos casos, a mensagem de *LABEL\_REQUEST* deverá ser propagada ao(s) nó(s) *downstream* até que se atinja o nó de egresso.

A informação no campo do contador de número de hops pode ser empregada por *LSR's* ATM que não suportam o campo TTL no rótulo, para que o campo *Time-to-Live* possa ser ajustado no cabeçalho IP de forma que, quando o datagrama deixe o domínio dos *LSR's* ATM, o valor do campo reflita o número de hops pelo qual o datagrama já passou.

O processo de supressão de uma associação de rótulo a uma FEC pode ser feito através de duas mensagens:

- *LABEL\_WITHDRAW*: enviada quando o *LSR* que distribuiu um rótulo não mais reconhece uma FEC que antigamente era conhecida, possivelmente devido a alguma modificação percebida pelo protocolo de roteamento;
- *LABEL\_RELEASE*: enviada nas seguintes situações:
  1. O *LSR* que fez o mapeamento do rótulo não é mais o próximo *hop* para aquela FEC e o *LSR* está operando no modo conservador de retenção de rótulo;
  2. O *LSR* que recebeu o rótulo percebe que o *LSR downstream* que distribuiu o rótulo não é mais o próximo nó para aquela FEC, opera no modo conservador de retenção de rótulo e decide solicitar a supressão daquele mapeamento;
  3. O *LSR* recebeu uma mensagem *LABEL\_WITHDRAW* para um rótulo que ele havia recebido anteriormente. Ele então envia *LABEL\_RELEASE*.

Se o *LSR* estiver operando no modo liberal de retenção de rótulo, os casos (1) e (2) acima não se aplicam. O *LSR upstream* manterá cada rótulo, ainda que não usado, para que ele possa ser imediatamente utilizado mais tarde se um outro *LSR downstream* tornar-se o próximo nó para aquela FEC.

A figura 2.12 ilustra algumas seqüências de mensagens LDP.



### Outras Mensagens

Outras mensagens definidas no protocolo LDP são: *LABEL\_QUERY*, *ADDRESS\_MESSAGE*, *ADDRESS\_WITHDRAW*, *EXPLICIT\_ROUTE\_REQUEST* e *EXPLICIT\_ROUTE\_RESPONSE*.

A mensagem *LABEL\_QUERY* é empregada nos procedimentos de detecção e prevenção de *loop* e será detalhada na seção específica que trata desta questão.

A mensagem *ADDRESS\_MESSAGE* é empregada para divulgar a lista de endereços IP alcançáveis pelas interfaces do LSR. Deve ser enviada sempre que uma sessão LDP é iniciada, antes de se enviar mensagens *LABEL\_MAPPING*. O anúncio dos endereços de interface pode ser feito com uma ou mais mensagens *ADDRESS\_MESSAGE*.

A mensagem *ADDRESS\_WITHDRAW* é empregada para retirar endereços anunciados anteriormente.

O uso da mensagem *ADDRESS\_MESSAGE* permite que roteadores de ingresso tomem conhecimento dos endereços IP alcançáveis pelos nós de egresso, mesmo que a rede em questão não empregue um roteamento de estado de enlace. Desta forma, os LSR's de ingresso podem fazer agregação do tipo "todos os pacotes que saem por um determinado LSR de egresso", o que corresponde a uma FEC do tipo "identidade do roteador" (de egresso).

As mensagens *EXPLICIT\_ROUTE\_REQUEST* e *EXPLICIT\_ROUTE\_RESPONSE* servem ao propósito de estabelecer LSP's empregando-se roteamento explícito, e serão detalhadas posteriormente quando discutirmos o uso do roteamento explícito pelo MPLS.

#### 2.3.2 Detecção de *loops*

Há situações transitórias na rede que podem levar à ocorrência temporária de *loops* de roteamento. *Loops* de roteamento podem surgir, por exemplo, devido a falhas nos enlaces entre roteadores, o que leva a mudanças nas tabelas de roteamento. Dependendo da ordem na qual os roteadores atualizam suas tabelas de roteamento em função das modificações ocorridas na rede, *loops* poderão existir durante algum tempo, até que todos os roteadores passem a refletir a nova realidade da topologia.

No caso de MPLS, caso ocorra o estabelecimento de um LSP que contenha um *loop*, os pacotes que ingressarem neste LSP circularão pelo *loop* até que o campo TTL do pacote seja zerado. Neste momento o pacote será retirado de circulação.

No entanto, no caso de LSR's ATM, estes não apresentam o campo TTL

como no caso do encapsulamento genérico. Ou seja, se não se impedir a formação do *loop*, os pacotes circularão indefinidamente pela rede, provocando um efeito desastroso em termos de tráfego.

A arquitetura MPLS provê técnicas para garantir que *loops* sejam detectados durante a criação de um LSP e técnicas estão em discussão para auxiliar na prevenção de *loops*. Estas técnicas, entretanto, não estão incluídas atualmente na especificação do MPLS.

O mecanismo de detecção de *loop* é obrigatório no protocolo LDP. O algoritmo de detecção de *loop* é executado à medida que se estabelece o LSP distribuindo-se a mensagem LABEL\_MAPPING sucessivamente de um LSR para outro. A mensagem LABEL\_MAPPING também contém o objeto path vector. Ao receber a mensagem, um LSR realiza detecção de *loop* verificando se seu endereço IP está presente no *path vector*. Se um *loop* for detectado, o LSR deverá transmitir um NAK ao nó que enviou o LABEL\_MAPPING e não instalar o novo rótulo recebido ou propagá-lo adiante. Além disso, se ele já tiver distribuído um rótulo *upstream* para aquela FEC, ele deverá retirá-lo. Caso não se detecte a existência de *loop* no *path vector*, o LSR deverá concatenar seu próprio endereço IP no *path vector* antes de propagar a mensagem LABEL\_MAPPING adiante.

### 2.3.3 Seleção de Rota

Rotas podem ser selecionadas de duas maneiras:

- Roteamento *hop-by-hop*: Cada nó independentemente escolhe o próximo nó para cada FEC. Um LSP roteado *hop-by-hop* é um LSP cujo caminho é definido usando roteamento *hop-by-hop*.
- Roteamento explícito: Um único LSR, normalmente o LSR de ingresso ou de egresso, especifica por quais LSR's o LSP deve passar. Estritamente explícito é quando um único LSR especifica o LSP inteiro. Fracamente explícito é quando um único LSR especifica apenas alguns dos LSR's no caminho.

A sequência de LSR's seguida por um LSP roteado explicitamente pode ser escolhida por configuração, ou pode ser selecionada dinamicamente por um único nó. Por exemplo, o nó de egresso pode fazer uso da informação topológica obtida a partir de um protocolo de estado de enlace a fim de determinar o caminho inteiro para a árvore terminando naquele nó,

Roteamento explícito pode ser útil para diversos propósitos, tais como políticas de roteamento ou engenharia de tráfego. No MPLS, a rota explícita

tem que ser especificada no momento em que o LSP é criado (rótulos atribuídos), não sendo necessário especificá-la para cada pacote. Isto torna o roteamento explícito do MPLS muito mais eficiente do que a alternativa de roteamento IP baseada na origem (*source routing*), onde cada pacote deve carregar a informação da rota toda.

### Roteamento Explícito com LDP

O grupo de trabalho de MPLS no IETF chegou recentemente a um consenso de que o LDP deve prover mecanismos para suportar roteamento explícito com provisão para indicação de prioridade.

O que se propõe é um mecanismo de estabelecimento de conexão fim-a-fim, que pode, em princípio, ser invocado por qualquer das pontas do LSP a ser explicitamente roteado. Apresenta-se nesta seção o mecanismo para o estabelecimento de um LSP explicitamente roteado a partir do nó de ingresso, discutido em [4]. Acredita-se que a latência associada a este mecanismo de estabelecimento fim-a-fim seja tolerável, uma vez que as aplicações que mais motivam o emprego de roteamento explícito, por exemplo Engenharia de Tráfego, geralmente implicam que os LSP's assim estabelecidos terão longa duração (relativamente ao tempo de estabelecimento da conexão).

Atualmente, a especificação do LDP [4] apenas especifica LSP explicitamente roteados para conexões ponto-a-ponto unidirecionais. O mecanismo, no entanto, pode ser estendido para acomodar LSP's multiponto-ponto.

Duas mensagens do protocolo LDP suportam o estabelecimento de LSP's explicitamente roteados: EXPLICIT\_ROUTE\_REQUEST e EXPLICIT\_ROUTE\_RESPONSE.

O processo de estabelecimento se inicia com o envio da mensagem EXPLICIT\_ROUTE\_REQUEST pelo nó de ingresso do LSP. Esta mensagem consiste de uma lista de objetos denominados FEC\_ER (*FEC-Explicitly Routed*) num formato do tipo TLV (*Type, Length, Value*). Cada FEC\_ER é constituído por outros objetos aninhados, conforme indicado na figura 2.13.

A mensagem EXPLICIT\_ROUTE\_REQUEST permite especificar uma classe de serviço a ser associada ao LSP, bem como atribuir níveis de preempção ao LSP para efeitos de Engenharia de Tráfego e Qualidade de Serviço. Em situações de falta de recursos na rede, LSP's com pouca prioridade poderão perder pacotes ou mesmo serem retirados. Também pode-se reservar recurso de largura de banda para o LSP.

A mensagem EXPLICIT\_ROUTE\_REQUEST é enviada ao primeiro nó da lista especificada no objeto EXPLICIT\_ROUTE. No primeiro nó, o objeto NEXT\_ERNH é incrementado, passando a apontar para o próximo nó, e

assim sucessivamente. No caso de roteamento estritamente explícito, quando a mensagem `EXPLICIT_ROUTE_REQUEST` é recebida em um nó, o `NEXT_ERNH` aponta para o ERNH que contém o endereço IP daquele nó.

No caso de roteamento estritamente explícito, um nó intermediário ao receber a mensagem `EXPLICIT_ROUTE_REQUEST` analisa o próximo objeto ERNH para determinar a interface por onde o LSP deverá ser estendido. Neste caso, o próximo nó deverá ser sempre adjacente.

Se o indicador apontar roteamento fracamente explícito naquele nó intermediário, então, dependendo do endereço IP presente no objeto ERNH seguinte, um próximo *hop* é selecionado com base na informação contida na FIB.

O nó intermediário processa as reservas de largura de banda (se houver alguma). Caso a reserva não possa ser atendida, uma resposta indicadora deste fato é retornada ao nó anterior. Na recepção da mensagem `EXPLICIT_ROUTE_REQUEST`, o recurso é apenas reservado. A largura de banda será efetivamente comprometida quando a resposta vier de volta do LSR de egresso.

Quando a mensagem `EXPLICIT_ROUTE_REQUEST` atingir o LSR especificado no último objeto ERNH e o pedido puder ser atendido, o LSR de egresso enviará uma mensagem `EXPLICIT_ROUTE_RESPONSE` indicando sucesso no estabelecimento do LSP. O nó de egresso incluirá um rótulo na mensagem de resposta.

A mensagem `EXPLICIT_ROUTE_RESPONSE` é enviada de volta pelo caminho reverso. Um nó intermediário ao receber tal resposta aloca um novo rótulo em sua interface de entrada e cria uma conexão entre o novo rótulo e o rótulo dado na mensagem. O LSR também faz o *commit* da largura de banda previamente reservada. O LSR então encaminha a mensagem a seu nó anterior com o novo rótulo. Quando a mensagem de resposta atingir o LSR de ingresso, o LSP explicitamente roteado é declarado em serviço.

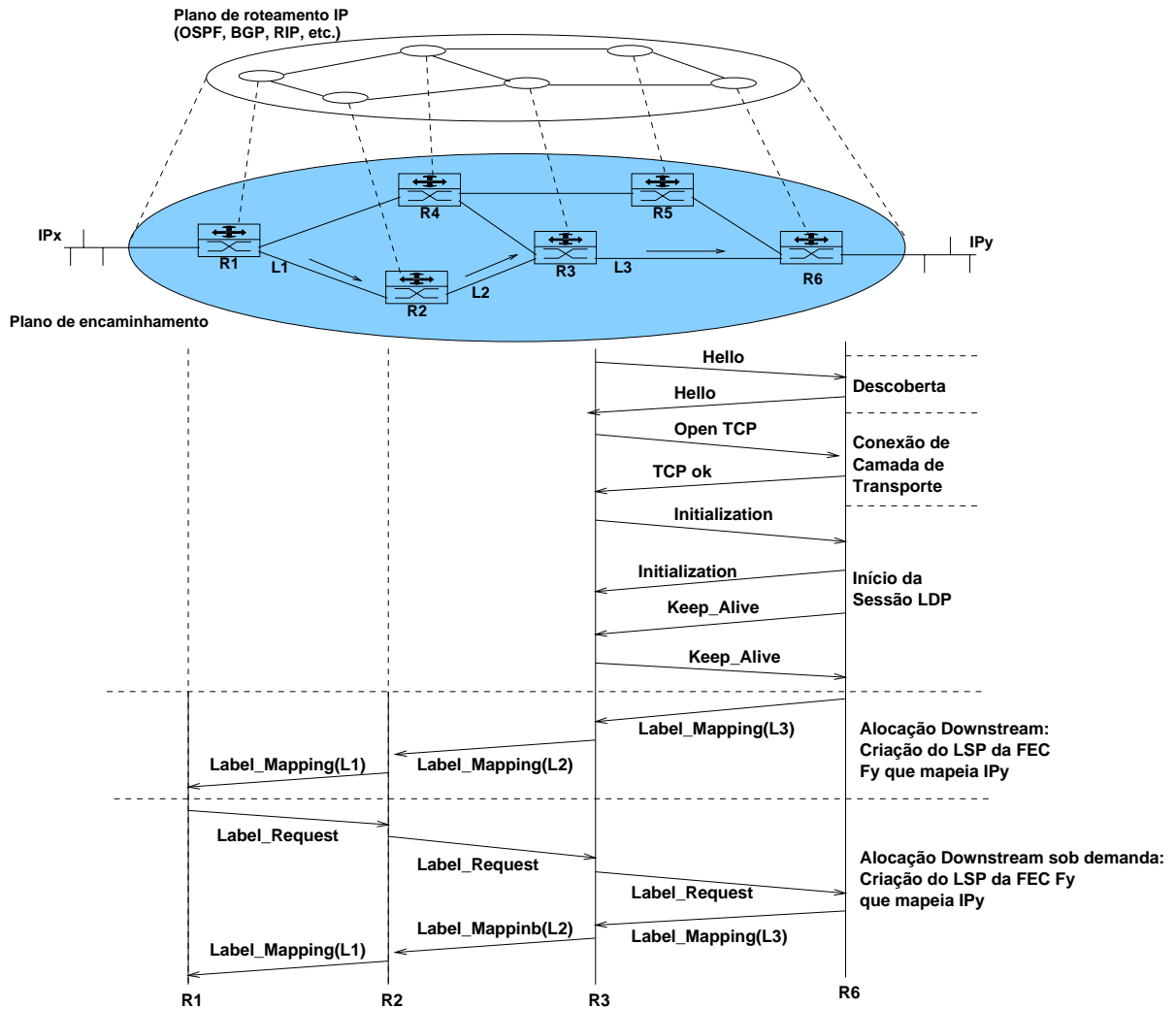


Figura 2.12: Exemplos de Mensagens LDP (*Alocação Downstream e Alocação Downstream sob Demanda*)

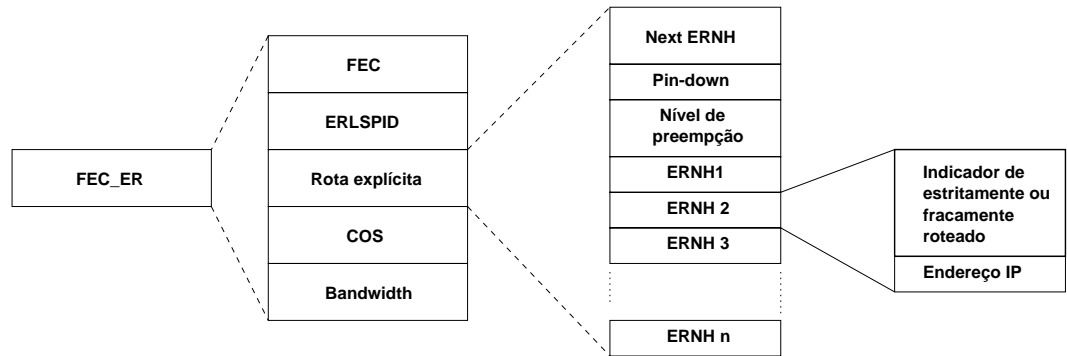


Figura 2.13: Roteamento explícito através do LDP