MODELOS DE INTERCONEXÃO DE DADOS SOBRE REDES ATM

Profs. Eleri Cardozo & Mauricio Magalhães
DCA/FEEC/UNICAMP

1 Introdução

Uma infra-estrutura de *chaves* 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 nuvem ATM, a qual se espera formada por *chaves* de fabricantes diversos, há 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-Fórum 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 rede 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 (PABXs) e de vídeo (CODECs). Estes serviços possuem características de tráfego distintas e, no entanto, devem ser suportados por uma mesma infra-estrutura ATM.

O interesse principal desta parte do 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 (*Lan Emulation* do ATM-Fórum), o IP-Clássico (IETF) e MPOA (*Multiprotocol Over ATM* – ATM-Forum). Estes 2 últimos modelos também podem ser vistos como soluções que possuem expansibilidade 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 do ATM podemos destacar soluções do tipo IP-*Switching*, *Tag-Switching*, IP-*Navigator* e outras, que surgem como alternativas às soluções propostas pelo IETF e ATM-Fórum.

1.1 Modelos ATM para interconexão de redes

A justificativa do desenvolvimento de modelos ATM para interconexão das redes atualmente existentes, deve-se ao fato de que a utilização inicial reservada para a tecnologia ATM foi a de *backbone* de alta velocidade para suportar o tráfego de dados entre redes locais e no transporte de dados em redes de longa distância .

Estes modelos podem ser classificados em dois tipos principais: modelo o*verlay* 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, no caso desta última, na utilização do mesmo esquema de endereçamento utilizado pelo protocolo da camada de rede. Neste caso, equipamentos conectados à rede ATM seriam identificados com endereços da camada de rede, por exemplo através de endereços IP, os quais seriam utilizados também pela sinalização ATM. Este tipo de modelo denomina-se modelo *peer*.

Uma outra possibilidade seria desacoplar a nuvem ATM de qualquer outro protocolo que venha a utilizar a rede ATM como rede de transporte. Isto implica em definir uma estrutura de endereçamento própria e protocolos de roteamento e sinalização para a rede ATM. No caso temos a coexistência de duas estruturas de endereçamento diferentes e, consequentemente, protocolos de roteamento e sinalização também diferentes, o que requer um mapeamento do endereço da camada superior (ex. endereço IP) em um endereço ATM na rede ATM. Este tipo de solução caracteriza o denominado modelo *overlay*. A grande vantagem deste modelo 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 transporte multiserviço, já que a nuvem ATM não fica vinculada a qualquer protocolo da camada superior.

O esforço tanto do IETF como do ATM-Fórum consiste em privilegiar o modelo o*verlay* 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 para o ATM, 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 P-NNI (*Private*-NNI), constituido 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.

1.2 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 intermediário deve ser entendido como um *switch* ATM enquanto um sistema final (*end-system*) pode ser um LAN *switch*, PC, estação de trabalho ou roteador. Este último é normalmente considerado nas redes tradicionais como um elemento intermediário no nível da camada de rede.

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 como nas redes locais. Caso a subrede seja incapaz de encaminhar pacotes *multicasting* sem a utilização de servidores adicionais, a subrede é denominada de NBMA (*Non-Broadcast Multiple Access*).

2 Modelo Overlay

Como discutido anteriormente, o modelo *overlay* caracteriza-se pela utilização de estruturas de endereçamento diferentes entre os protocolos superiores e a infra-estrutura ATM. No caso da camada de rede os protocolos 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-Fórum especificou o protocolo PNNI onde o roteamento entre *chaves* ATM é uma das funções principais, consequentemente, 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 associar a rede de transporte ATM a uma estrutura no nível de enlace como, por exemplo, uma rede Ethernet. No caso desta última, o endereçamento da camada de enlace é baseado em endereços atribuidos 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. Consequentemente, o projetista de rede não possui controle sobre a alocação do endereço MAC, com exceção do DECnet que permite a redefinição de endereços MAC. Neste último caso, os endereços de rede DECnet podem ser derivados a partir dos endereços MAC definidos. Consequentemente, um dos problemas básicos no caso dos 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 *broadcasting* 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 de *broadcast*. Neste caso é necessário o emprego de servidores para resolução de endereços como teremos oportunidade de discutir nos itens a seguir onde serão apresentadas as propostas mais importantes que empregam o modelo *overlay*.

2.1 Modelo Clássico (IETF)[1]

Os documentos básicos relativos ao Modelo Clássico são *Classical IP and ARP over ATM* [RFC 1577] e *Classical IP and ARP over ATM update*. Estas RFCs especificam

[1] Baseado na referência [Alb96].

a interação entre sistemas ATM (*end-systems*) através do encapsulamento dos dados conforme especificado no documento *Multiprotocol Encapsulation over ATM Adaptation Layer 5* [RFC 1483] e o mecanismo de resolução de endereço IP no correspondente endereço ATM.
A Figura 1 ilustra o contexto para utilização do Modelo Clássico. Inicialmente temos uma rede corporativa TCP/IP, caracterizada por um *backbone* baseado em rede Ethernet, Token-Ring ou FDDI e utilizado para interconexão de redes locais de departamentos através da utilização de roteadores. Cada rede departamental e o *backbone* são subredes IP.

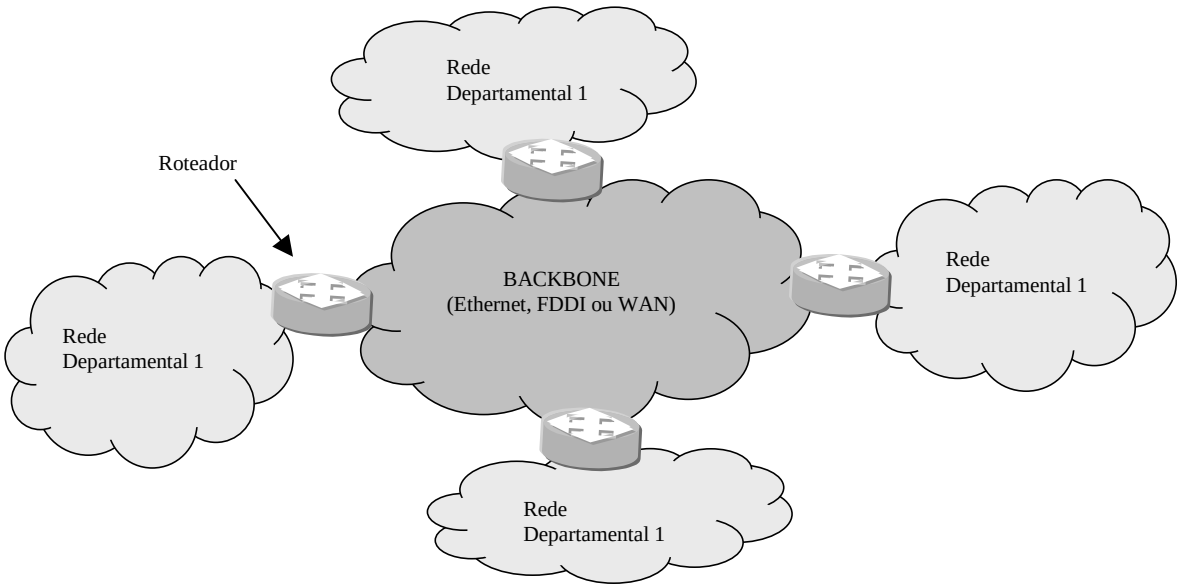


Figura 1: Exemplo de uma rede TCP/IP corporativa.

Em um dado instante será necessário aumentar a capacidade do tráfego do *backbone* da Figura 1, em função do aumento da carga nas subredes departamentais através do uso de novos serviços com caracteríticas multimídia e acesso aos servidores departamentais. Esta substituição do *backbone* ocorrerá através da utilização de uma nuvem ATM que possibilitará a conectividade em alto desempenho dos roteadores conectados às redes departamentais.

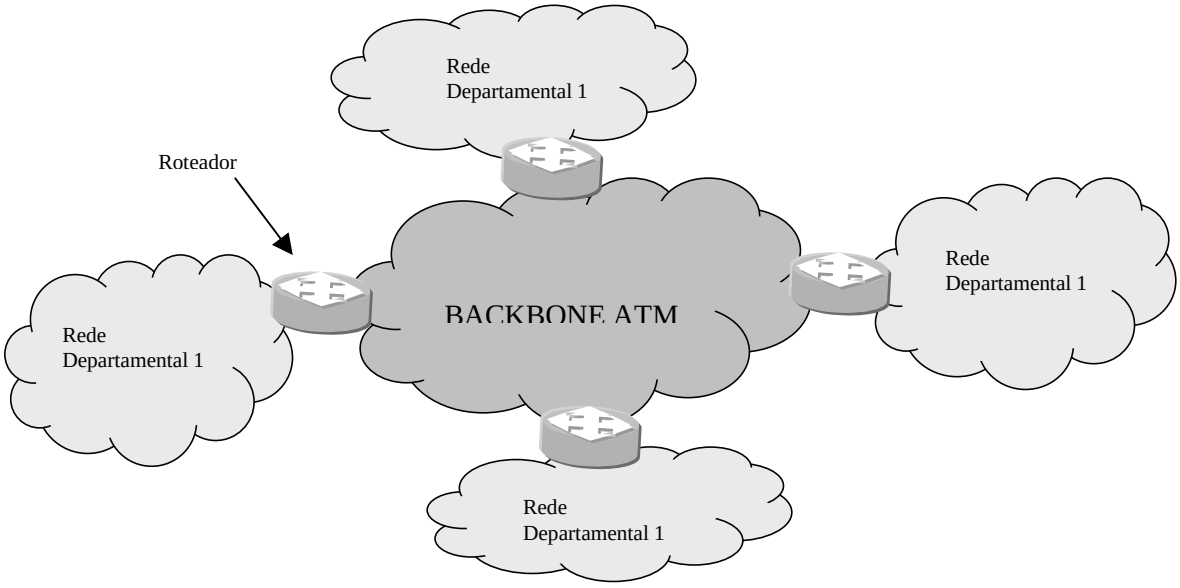


Figura 2: Rede Corporativa com *Backbone* ATM.

A utilização do IP-Clássico na estrutura representada pela Figura 2 possui extensões ao modelo clássico original que permite o uso de soluções para o fornecimento de *broadcasting/multicasting* através de uma rede ATM, como também extensões ao roteamento ATM, este último baseado no protocolo NHRP (*Next Hop Resolution Protocol*). Estas extensões são fundamentais para proporcionar escalabilidade ao Modelo Clássico.

2.1.1 Encapsulamento Multiprotocolo sobre o ATM AAL5[2]

O IETF trabalhou primeiramente na definição de métodos que permitissem o transporte de múltiplos tipos de protocolos da camada de rede ou de enlace através de uma conexão AAL5 (*ATM Adaptation Layer Type 5*), e também na multiplexação de vários tipos de protocolos sobre uma mesma conexão.
Dois métodos de encapsulamento para carregar tráfego de interconexão de rede sobre ATM AAL5 foram definidos na RFC-*Request For Comments* 1483 [RFC 1483]:

Encapsulamento LLC/SNAP: Neste método, múltiplos tipos de protocolos podem ser carregados através de uma única conexão virtual, com o tipo do pacote encapsulado identificado por um cabeçalho IEEE 802.2 LLC/SNAP (Logical Link Control/SubNetwork Attachment Point). O método é semelhante ao utilizado pelo IEEE 802.2 e SMDS (Switched Multimegabit Data Service);

[2] Baseado na RFC 1483.

Multiplexação Baseada em VC: No método de multiplexação de conexão virtual (VC-Virtual Connection), apenas um protocolo é carregado através de uma conexão ATM, com o tipo do protocolo implicitamente identificado no estabelecimento da conexão. Como resultado, nenhum campo de multiplexação ou de tipo de pacote é necessário.
Quando a RFC foi originalmente apresentada, estimava-se que o método baseado na multiplexação de conexão virtual seria mais comum nos ambientes locais onde a comutação de circuito fosse suportada (*SVC-Switched Virtual Connection*), pelo fato dos custos das conexões virtuais não serem uma questão relevante nestes ambientes. Por outro lado, esperava-se que a multiplexação baseada no encapsulamento LLC/SNAP fosse de maior interesse nos ambientes de longa distância, onde os custos associados aos canais virtuais são mais importantes. Na realidade, a maior parte das implementações nos ambientes locais ou de longa distância empregam o método de encapsulamento LLC/SNAP. A escolha do método de multiplexação pode ser implícito no caso das redes baseadas em canais virtuais permanentes (*PVCs – Permanent Virtual Channels*), ou configurado pelo gerenciamento da rede, ou através da sinalização no caso dos ambientes baseados em SVCs.

2.1.2 Seleção do Método de Multiplexação

Segundo [Alles], o encapsulamento LLC/SNAP é o método de encapsulamento mais utilizado nos protocolos IP sobre ATM. O ITU-T e o grupo de trabalho *ATM Forum Multiprotocol Over ATM* estão utilizando este método como *default* para o transporte de multiprotocolos sobre ATM.

2.1.2.1 Encapsulamento LLC/SNAP

Este método de encapsulamento é utilizado quando vários protocolos são carregados sobre o mesmo VC. Para permitir ao receptor processar adequadamente o AAL5 CPCS-PDU (*Common Part Convergence Sublayer-Packet Data Unit*), o campo de dados (*Payload Field)* do CPCS-PDU deve conter informações necessárias para identificar o protocolo da PDU roteada.
No Encapsulamento LLC/SNAP o protocolo da PDU roteada é identificado por um cabeçalho IEEE 802.2 LLC seguido de um cabeçalho SNAP. O cabeçalho LLC é composto de 3 octetos, e o cabeçalho SNAP é composto de 5 octetos.

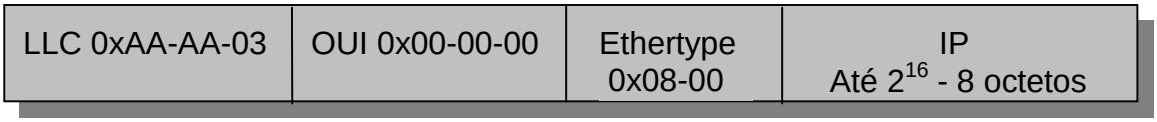


Figura 3: Encapsulamento LLC/SNAP para protocolo IP.

onde:
LLC = 0xAA-AA-03 indica a presença de um cabeçalho SNAP, que é composto dos campos OUI e PID;
OUI (Organizationally Unique Identifier) = 0x00-00-00 indica que o tipo é um Ethertype[3];
Ethertype = 0x08-00 indica tratar-se de uma PDU IP.

A Figura 4 mostra um exemplo de encapsulamento de um datagrama IP em um CPCS-PDU da camada de adaptação AAL5, e a transmissão em células ATM. Inicialmente o transmissor estabelece um SVC (*Switched Virtual Circuit*) ou um PVC (*Permanent Virtual Circuit*) através da rede ATM com o *host* de destino, e especifica que o circuito deve utilizar a AAL5. Quando do envio do datagrama este é então passado para a camadade de adaptação AAL5. A AAL5 gera um *trailer* no final do CPCS-PDU *payload*, divide o datagrama em células e transfere as células através da rede. No receptor, a AAL5 remonta o datagrama, usa a informação no *trailer* para verificar se algum *bit* foi perdido ou corrompido, e passa o datagrama encapsulado no *payload* da AAL5 para o protocolo IP.

2.1.2.2 Multiplexação Baseada em VC

Neste método de encapsulamento, o protocolo de interconexão de rede carregado é identificado implicitamente pela conexão VC entre duas estações ATM. Não há portanto a necessidade de incluir campos de informação de multiplexação junto ao *payload* do AAL5 CPCS-PDU.

2.1.3 Unidade Máxima de Transferência (MTU) do IP sobre a AAL5

Várias aplicações sobre TCP (*Transmition Control Protocol*) indicam o tamanho da unidade máxima de transmissão (MTU-*Maximum Transmission Unity*) para o IP (*Internet Protocol*) visando melhor desempenho. Como a fragmentação de datagramas IP é tida como altamente indesejável, o tamanho máximo da unidade de transmissão (MTU) para IP sobre ATM AAL5 deve ser razoavelmente grande.
A [RFC 1209] especifica a MTU do IP sobre SMDS como 9180 octetos, e como não existe uma boa razão para este valor não ser utilizado como *default* da MTU para IP sobre ATM, este foi também adotado, de acordo com a [RFC 1626]. O uso do mesmo valor para ATM e SMDS tem como objetivo facilitar a interoperabilidade entre estes serviços.
Quando um datagrama IP é maior que a MTU da rede, o IP fragmenta o datagrama e passa cada fragmento para a AAL5. Embora a camada AAL5 suporte CPCS-PDU

[3] Ethertype: permite identificar os vários protocolos de rede (IP, IPX, ARP) e enlace (IEEE 802.3, etc.).

Payloads de até 64k octetos, não existe nenhuma restrição utilizar-se valores menores, como 9180 octetos.

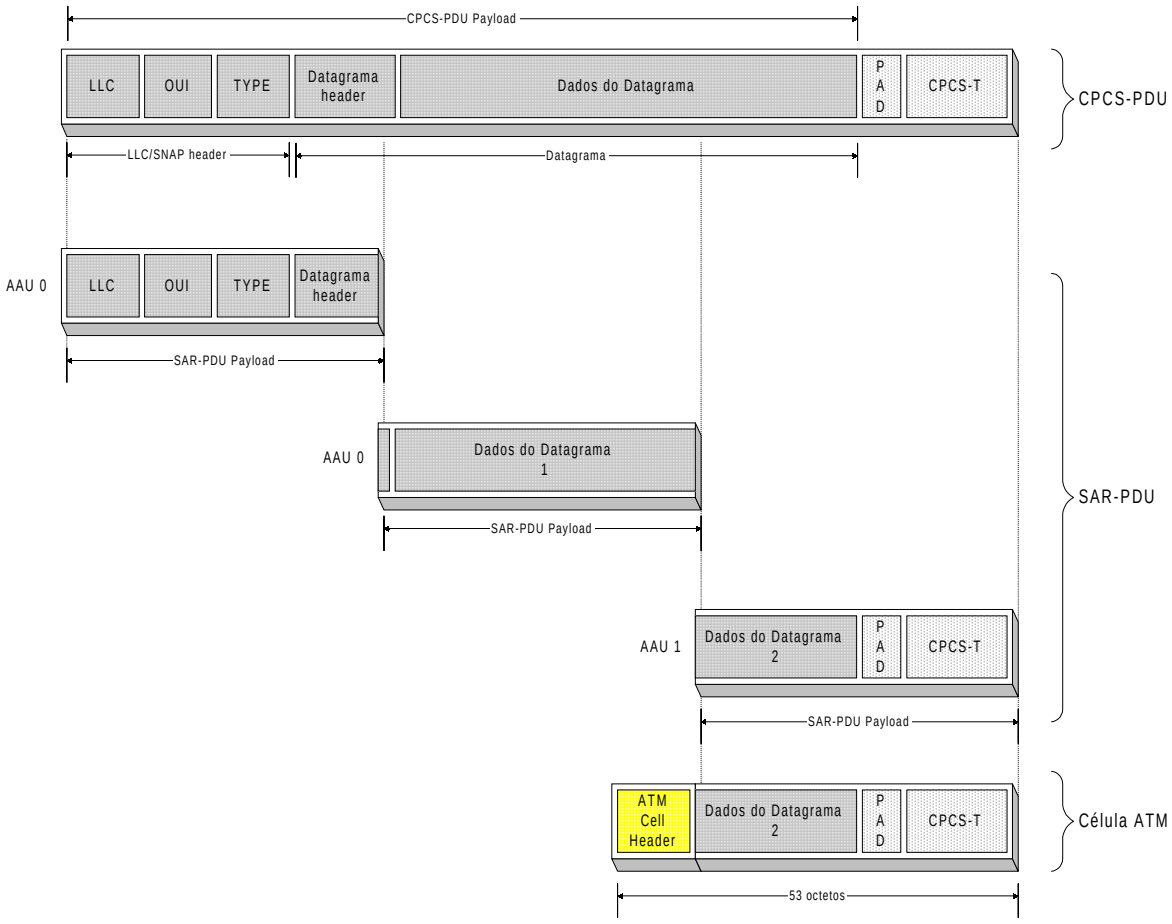


Figura 4: Exemplo de Encapsulamento LLC/SNAP sobre ATM AAL5.

Implentações que apenas suportam PVCs não utilizam nenhum protocolo de sinalização ATM, e portanto utilizam a MTU *default* de 9180 octetos, a menos que ambas as partes da conexão estejam de acordo com outros valores.
Implementações que suportam SVCs podem negociar o tamanho da AAL CPCS-PDU utilizando um protocolo de sinalização ATM. O protocolo de sinalização ATM usa duas diferentes partes de um IE (*Information Element*) chamado "parâmetros AAL" para trocar informações da MTU sobre o circuito ATM que está sendo estabelecido. Um campo *Forward Maximum CPCS-PDU size* contém o tamanho do MTU no sentido direto, e um campo *Backward Maximum CPCS-PDU size* contém o tamanho do MTU no sentido reverso.

2.1.4 Resolução de Endereço

O outro aspecto importante no modelo *overlay*, além da solução para o encapsulamento multiprotocolo em uma rede multiserviço, diz respeito à resolução de endereços pois temos duas estruturas de endereçamento que precisam ser relacionadas.

No caso do IP-Clássico a nuvem ATM pode ser constituida de uma LIS (*Logical IP Subnet*) ou múltiplas LISs. O conceito de LIS assemelha-se à forma como o IP enxerga uma subrede, ou seja, uma LIS corresponde a um conjunto de endereços de *hosts* resumidos em um prefixo de subrede. O IP-Clássico requer a utilização de roteador para a interação de membros de LISs diferentes, o emprego do encapsulamento LLC/SNAP como definido na RFC 1483 e limita o tamanho máximo de quadro em 9180 bytes. Deve ser observado que mesmo sendo possível a comunicação através de VCC entre dois *hosts* em LISs diferentes conectados através da nuvem ATM, o Modelo Clássico não permite esta interação direta.

Para realizar o transporte do pacote IP na rede ATM, um mecanismo deve ser usado para fazer a resolução dos endereços IP em endereços correspondentes ATM. Por exemplo, considere o caso de dois roteadores conectados através de uma rede ATM. Se um dos roteadores recebe um pacote através de uma interface LAN (*Local Area Network*), primeiramente ele verificará sua tabela de roteamento para determinar através de qual porta, e para qual roteador, ele deverá enviar o pacote. Se esta verificação indicar que o pacote deve ser enviado através de uma interface ATM, o roteador então precisa de um mecanismo que permita determinar o endereço ATM do roteador destino. O encapsulamento de um datagrama para transmissão através de uma rede ATM é de certa forma simples, porém a resolução de endereços IP em endereços ATM pode ser bastante díficil. Ao contrário de outras tecnologias, o ATM designa para cada computador conectado à rede um endereço ATM que deve ser utilizado quando se estabelece um circuito virtual. Por outro lado, como um endereço ATM é maior que um endereço IP, um endereço ATM não pode ser codificado dentro de um endereço IP, o que dificulta o relacionamento entre as duas estruturas de endereçamento. Por outro lado, o *hardware* ATM não suporta *broadcast* . Assim, o IP não pode usar o ARP convencional para fazer a resolução de endereços em redes ATM.

As conexões virtuais permanentes ATM (PVCs) complicam ainda mais a resolução de endereços IP. Uma vez que o gerente da rede deve configurar cada PVC manualmente, um *host* apenas conhece o par de identificadores de circuito VPI/VCI. Uma aplicação no host pode não conhecer o endereço IP ou o endereço ATM do destino. Desta forma, um mecanismo de resolução de endereços IP em rede ATM deve prover a identificação de um computador remoto conectado a um PVC, bem como a criação dinâmica de SVCs para destinos correspondentes a endereços ATM conhecidos.

Como o ATM é uma tecnologia orientada a conexão há a necessidade de dois níveis de resolução envolvendo endereços e conexões virtuais. Primeiro, quando se cria uma conexão virtual sobre a qual os datagramas serão enviados, o endereço IP do destino deve ser mapeado para o endereço do dispositivo conectado na rede ATM. De posse deste endereço é possível criar-se um circuito virtual. Segundo, quando se envia um datagrama para um computador remoto sobre um circuito virtual existente, o endereço de destino IP deve ser mapeado para o par VPI/VCI do circuito. A

segunda ligação é usada cada vez que um datagrama é enviado sobre uma rede ATM. A primeira ligação é necessária apenas quando o *host* cria um SVC com o destino.
O grupo de trabalho do IP sobre ATM do IETF definiu um protocolo para suporte de resolução automática de endereços IP na [RFC 1577]. Este protocolo é conhecido como "*Classical IP over ATM* " e introduz a noção de subrede lógica IP ou LIS (*Logical IP Subnet*) conforme discutido anteriormente.

2.1.5 Protocolo *IP-Clássico sobre ATM*

O propósito da RFC 1577 é permitir implementações compatíveis e interoperáveis para a transmissão de datagramas IP, requisições e respostas *ATM Address Resolution Protocol* (ATMARP) sobre a camada de adaptação ATM AAL5.
protocolo IP-Clássico foi proposto para uma situação onde um grupo de computadores usa uma rede ATM em lugar de uma rede local. Este grupo forma uma sub-rede lógica IP ou LIS (Logical IP Subnet).
Múltiplas redes lógicas IP podem ser definidas ao longo de um conjunto de computadores que estejam conectados na mesma rede ATM. A Figura 5 ilustra oito computadores conectados a uma rede ATM e divididos em duas LISs.

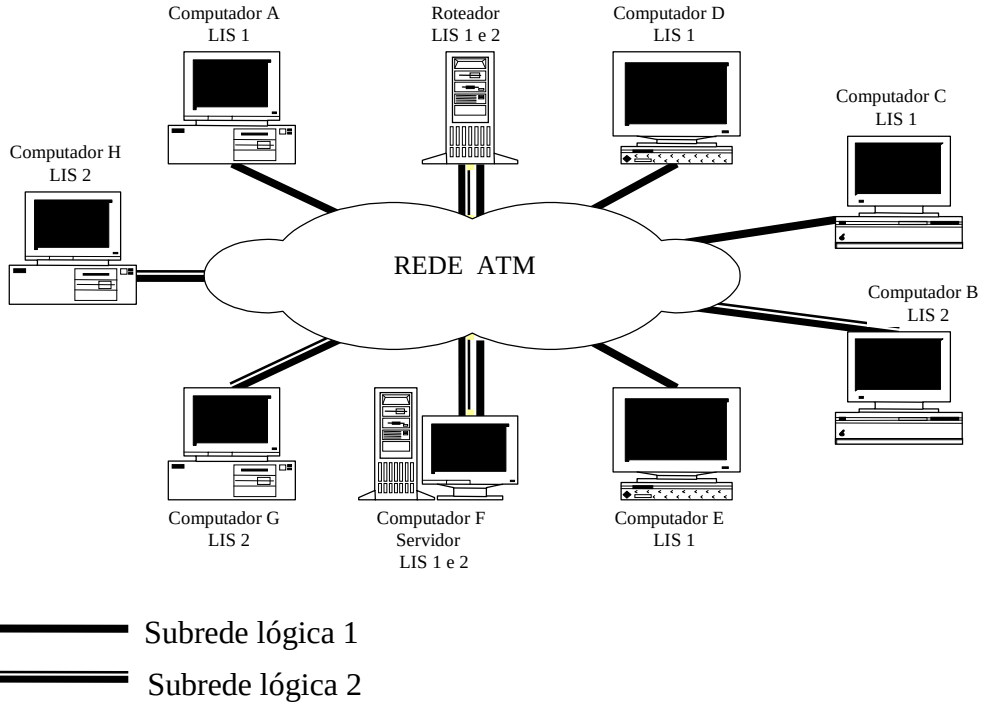


Figura 5: Exemplo do IP Clássico sobre ATM.

Todos os computadores estão conectados na mesma rede ATM. Os computadores A, C, D, E e F participam da LIS 1, enquanto os computadores B, F, G e H participam

da LIS 2. Cada LIS funciona como uma LAN separada. Os computadores participando de uma LIS podem estabelecer circuitos virtuais entre eles para troca de datagramas. Como uma LIS forma conceitualmente um rede separada os computadores em uma mesma LIS compartilham um mesmo prefixo de subrede IP, e este prefixo difere daquele utilizado por outras subredes lógicas IP.

Embora os computadores de uma LIS possam escolher uma MTU não padronizado, todos os computadores devem usar a mesma MTU em todos os circuitos virtuais que fazem parte da LIS.

Conforme mencionado anteriormente, um *host* em uma LIS não pode se conectar diretamente com um *host* em outra LIS. Em vez disto, todas as comunicações entre sub-redes lógicas IP devem ser feitas através de um roteador que participa de múltiplas sub-redes lógicas conforme no caso do roteador mostrado na Figura 5, o qual participa das duas subredes lógicas do exemplo através de uma mesma interface ATM. Os endereços ATM da estação e do servidor ATMARP devem ser configurados em cada estação IP conectada à rede ATM.

Em um ambiente SVC, as requisições ATMARP são enviadas para esse endereço e a conversão do endereço destino IP em endereço ATM é realizada por um servidor. Este servidor deve atender requisições ATMARP de todos os membros IP dentro de uma LIS.

Roteadores que desejam suportam a interconexão de várias LISs devem ser capazes de suportar múltiplos conjuntos de parâmetros e associar cada conjunto de parâmetros a um número de rede/sub-rede IP específico. Recomenda-se ainda que o roteador ofereça este suporte para múltiplas LISs com uma única interface ATM física, que pode ter um ou mais endereços ATM finais (*ATM Endpoint Address*).

2.1.5.1 ATMARP

A resolução de endereço dentro de uma sub-rede lógica IP faz uso de um protocolo de resolução de endereço ATM *(ATM Address Resolution Protocol - ATMARP*), bem como do protocolo de resolução de endereço ATM inverso (InATMARP). ATMARP é o mesmo protocolo ARP [RFC 826] com as extensões necessárias para suportar ARP em um ambiente de servidor ATM *unicast.* InATMARP é o mesmo protocolo original InARP, só que aplicado a redes ATM.

Como no ARP convencional, o transmissor forma uma requisição que inclui o seu endereço IP e o seu endereço ATM, bem como o endereço IP do destinatário cujo endereço ATM quer determinar. O transmissor envia então a requisição para o servidor ATMARP da sub-rede lógica. Se o servidor conhece o endereço ATM do destino, ele envia uma resposta ATMARP. Caso contrário, ele envia um resposta negativa ATMARP.

A resolução de endereços IP em sistemas orientados à conexão é um pouco mais complexo do que em ambientes não orientados à conexão. Como a nuvem ATM pode suportar os dois tipos de circuitos virtuais (PVC e SVC), dois casos devem ser abordados. Inicialmente consideraremos o caso de circuitos virtuais permanentes, em seguida será discutido o caso envolvendo circuitos virtuais comutados.

Utilizando Circuitos Virtuais Permanentes PVCs

Para entender que problemas os PVCs introduzem, lembramo-nos de como o *hardware* ATM trabalha. O administrador da rede deve configurar cada PVC; os *hosts* por si próprios não estabelecem tais conexões. Um *host* inicia a sua operação com os PVCs já configurados e não recebe qualquer informação de seu *hardware* ATM a respeito do endereço de algum ponto remoto. Assim, a menos que a informação de endereçamento tenha sido configurada em cada *host* (por exemplo, armazenada em disco), os hosts não sabem o endereço IP ou ATM do computador que é conectado via um PVC.

O protocolo InATMARP resolve o problema de encontrar endereços quando se usa PVCs. Para usar o protocolo a estação deve conhecer cada um dos PVCs que foram configurados. Para determinar o endereço IP e ATM de um ponto remoto, o computador deve mandar um pacote de requisição InATMARP com o campo OPERATION contendo o valor 8. Sempre que tal requisição é recebida em um PVC, o receptor gera uma resposta InATMARP com o campo OPERATION contendo o valor 9. Tanto a requisição como a resposta InATMARP contém o endereço IP e ATM do transmissor. Assim, o computador em um dos extremo da conexão aprende a ligação para o computador no outro extremo da conexão. Em síntese, dois computadores que se comunicam via um PVC usam o InATMARP para descobrir seus endereços IP e ATM.

Utilizando Circuitos Virtuais Comutados – SVCs

Dentro de uma LIS, computadores criam circuitos virtuais comutados sob demanda. Quando um computador A necessita enviar um datagrama para um computador B e nenhum circuito virtual existe para B, o computador A deve utilizar a sinalização ATM para criar o circuito virtual necessário. Como já comentado em outras partes deste texto, cada LIS tem um servidor ATMARP e todos os computadores de uma LIS devem ser configurados tal que eles saibam encontrar o seu servidor (por exemplo, um computador pode ter um PVC para o servidor ou pode ter o endereço ATM do servidor armazenado em disco). Um servidor não forma conexões para outros computadores, ele simplesmente espera por um contato dos computadores de uma LIS. O servidor ATMARP, antes de completar a conexão de um novo SVC deve especificar o encapsulamento utilizado como sendo o LLC/SNAP.

Para mapear o endereço IP do computador B em um endereço ATM, o computador A deve ter um circuito virtual aberto para o servidor ATMARP da sua LIS. O computador A forma um pacote de requisição ATMARP e envia-o ao servidor através da conexão estabelecida. O campo OPERATION do pacote contém o valor 1 e o endereço de destino do pacote contém o endereço IP do computador B.

Um servidor ATMARP mantém uma base de dados dos mapeamentos IP para ATM já realizados. Se o servidor conhece o endereço ATM do computador B, o protocolo ATMARP opera de maneira similar a um *proxy* ARP. O servidor forma uma resposta

ATMARP com o campo OPERATION indicando o valor 2 e preenchendo o campo de endereço destino ATM com o endereço ATM da estação B.
Se o servidor não conhece o endereço ATM que corresponde ao endereço destino IP, o ATMARP difere do ARP convencional. Ao invés de ignorar a requisição, o servidor envia um pacote de *negative acknowledgement* (um pacote ATMARP com o campo OPERATION igual a 10). O *negative acknowledgement* serve para distinguir entre um endereço IP não conhecido e um mal funcionamento do servidor. Assim, quando um *host* envia uma requisição ao servidor ATMARP, três situações podem ocorrer: O host pode receber o endereço ATM do destino, o destino não está atualmente disponível na LIS ou o servidor não está respondendo à requisição.
Um servidor ATMARP pode ser capaz de servir a diversas LISs. Neste caso ele deve receber um endereço IP referente a cada LIS para a qual ele atue como servidor ATMARP.

2.1.5.2 Características Operacionais do Cliente ATMARP

O cliente de um servidor ATMARP é responsável por contatar o servidor a fim de registrar sua própria informação ATMARP, consultar e atualizar a sua informação referente a outros membros IP. Isto indica que os clientes ATMARP devem ser configurados com o endereço ATM do servidor ATMARP. Além disto os clientes ATMARP devem:

- Iniciar a conexão VC para o servidor ATMARP a fim de transmitir e receber pacotes ATMARP e InATMARP;
- Responder a pacotes de requisições InARP_REQUEST;
- Gerar e transmitir pacotes de requisição ARP_REQUEST para o servidor ATMARP e processar pacotes ARP_REPLY e ARP_NAK do servidor apropriadamente. Os pacotes ARP_REPLY podem ser utilizados para construir/destruir informações sobre o cliente na tabela ATMARP;
- Gerar e transmitir pacotes InARP_REQUEST quando necessário e processar pacotes InARP_REPLY apropriadamente. Os pacotes InARP_REPLY podem ser utilizados para construir/destruir informações sobre o cliente na tabela ATMARP.

2.1.5.3 Formato do Pacote ATMARP/InATMARP

Endereços IP são designados independentemente de endereços ATM. Cada implementação de *host* deve saber seus próprios endereços IP e ATM, e deve responder a requisições de conversão de endereço apropriadamente. Os membros IP devem também usar ATMARP e InATMARP para converter endereços IP em endereços ATM quando necessário.
Os protocolos ATMARP e InATMARP usam o mesmo tipo de *hardware*, protocolo e código de operação. A localização destes campos dentro de um pacote ATMARP é a mesma encontrada nos pacotes ARP e InARP. O ATMARP faz uso de um código adicional de operação para ARP_NAK.

A Figura 6 apresenta o formato correspondente aos pacotes ATMARP e InATMARP, para os quais os respectivos campos estão especificados na sequência:

Hardware Type → (16 bits) - tipo de hardware. **0x0013** indica ATM.
Protocol Type → (16 bits) - tipo de protocolo carregado sobre ATM. **0x0800** indica protocolo IP.
SEND.HLEN → (8 bits) – tipo e tamanho do endereço ATM da origem (ver NOTA ao final da lista);
SEND.HLEN2 → (8 bits) – tipo e tamanho do sub-endereço ATM de da origem (ver NOTA ao final da lista);
OPERATION → (16 bits) – campo de código da operação (decimal):

⇒ ARP_REQUEST = 1
⇒ ARP_REPLY = 2
⇒ InARP_REQUEST = 8
⇒ InARP_REPLY = 9
⇒ ARP_NAK = 10

SEND.PLEN → (8 bits) - tamanho em octetos do endereço do protocolo de origem. Para IP este campo vale 4.
TAR.HLEN → (8 bits) - tipo e tamanho do endereço ATM do destino (ver NOTA ao final da lista);
TAR.HLEN2 → (8 bits) - tipo e tamanho do sub-endereço ATM do destino (ver NOTA ao final da lista);
TAR.PLEN → (8 bits) - tamanho em octetos do endereço do protocolo de destino. Para IP este campo vale 4;
Endereço ATM da Fonte → E.164 ou Fórum ATM NSAP (Network Service Access Point Address);
Subendereço ATM da Fonte → Fórum ATM NSAP;
Endereço IP da origem
Endereço ATM do destino → E.164 ou Fórum ATM NSAP
Subendereço ATM do destino → Fórum ATM NSAP
Endereço IP do destino

NOTA.: Os 8 bits destes campos, são subdivididos conforme mostra a Figura 7, onde:
bit 8 → Reservado para uso futuro (default = 0)
bit 7 → Tipo de endereço: 0 indica formato Fórum ATM NSAP e 1 indica formato E.164.
bits 6-1 → Tamanho do endereço em octetos. (MSB= bit 6, LSB=bit 0)

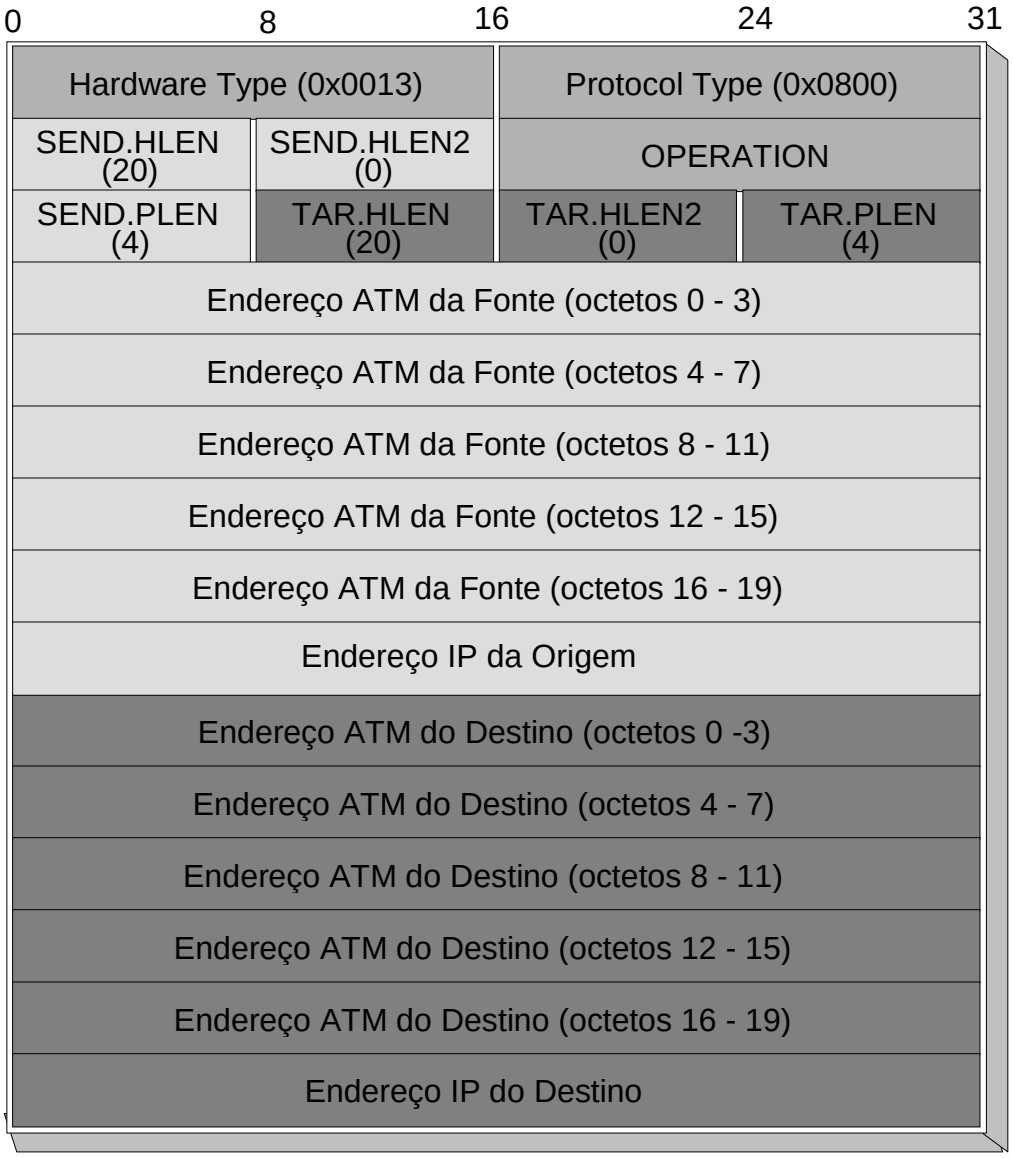


Figura 6: Formato de Pacote ATMARP e InATMARP.

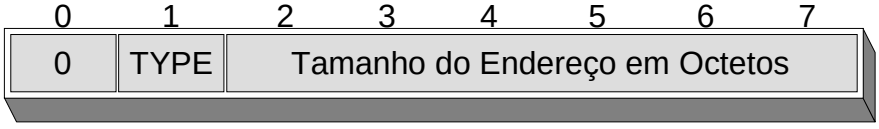


Figura 7: Subdivisão do campo *Target Protocolo Address.*

Segundo a [RFC 1577] os pacotes ATMARP e InATMARP devem ser carregados em um PDU da camada de adaptação AAL5 usando encapsulamento LLC/SNAP. O Formato do *payload* AAL5 CPCS-PDU para PDUs ATMARP e InATMARP é mostrado na Figura 8, onde:
O valor LLC igual a 0xAA-AA-03 (3 octetos) indica a presença de um cabeçalho SNAP;
O valor OUI igual a 0x00-00-00 (3 octetos) indica que os seguintes 2 bytes são referentes ao Ethertype;
O Ethertype igual a 0x08-06 (2 octetos) indica se tratar do encapsulamento de um pacote do protocolo ARP.

2.1.5.4 Subendereçamento

Os endereços ATM apresentados na [Q.93B] (como definido pelo Fórum ATM na especificação de sinalização [UNI 3.1]) incluem um Elemento de Informação *Calling Party Number* e um Elemento de Informação *Calling Party Subaddress.* Estes Elementos de Informação (IEs) contém o endereço e o subendereço da fonte ATM ATMARP/InATMARP, respectivamente. O Fórum ATM também definiu os Elementos de Informação: *Called Party Number* e *Called Party Subaddress*. Estes IEs contêm o endereço e o subendereço do destino ATM ATMARP/InATMARP, respectivamente.
O Fórum ATM, na [UNI 3.0], definiu três tipos de estruturas para o uso combinado do endereço e do subendereço ATM, conforme mostra a Tabela 1.

ESTRUTURA	Endereço ATM	Subendereço ATM
Estrutura 1	Fórum ATM NSAP	nulo
Estrutura 2	E.164	nulo
Estrutura 3	E.164	Fórum ATM NSAP

Tabela 1: Estruturas para uso combinado do endereço e do subendereço ATM.

Os membros IP devem registrar seus endereços ATM (*ATM Endpoint*) no seu servidor ATMARP usando a estrutura de endereço apropriada para a sua conexão ATM, isto é, uma LIS implementada sobre uma LAN ATM deve se registrar usando a estrutura 1. Uma LIS implementada sobre uma rede ATM pública que utiliza o E.164 deve se registrar usando a estrutura 2; Uma LIS implementada sobre uma combinação de LAN ATM e rede pública ATM deve se registrar usando a estrutura 3. Implementações basedas na [RFC 1577] devem suportar os três tipos de estruturas de endereçamento.

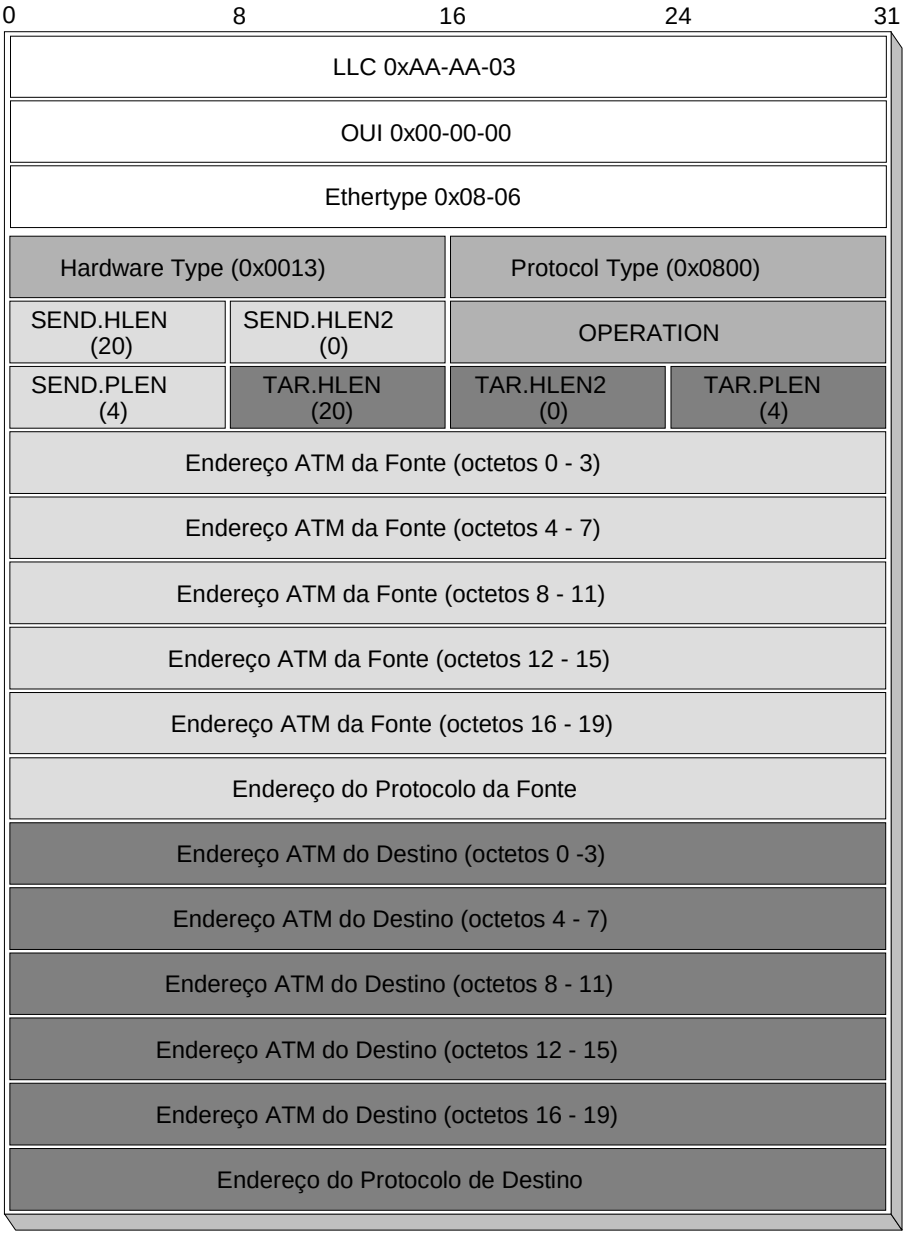


Figura 8: Formato do *payload* para PDUs ATMARP/InATMARP.

2.1.5.5 Tabela do servidor ATMARP

Um servidor ATMARP constrói e mantém sua base de dados de associações automaticamente. Para fazer isto, ele usa o InATMARP. Sempre que um roteador ou *host* abre um circuito virtual para o servidor ATMARP, o servidor imediatamente envia um pacote de requisição InATMARP. O host ou o roteador deve responder

com um pacote de resposta InATMARP. Quando o servidor recebe uma resposta InATMARP, o servidor extrai o endereço IP e ATM, e armazena a informação da associação em sua base de dados, ou seja, gera uma entrada em sua tabela ATMARP. Assim cada computador em uma LIS deve estabelecer uma conexão para o servidor ATMARP, mesmo se não for consultar o servidor.

2.1.5.6 Tempo de Vida das Informações ATMARP

Assim como as ligações de endereços feitas pelo ARP convencional, as associações feitas via ATMARP devem ter um tempo de vida especificado e após este tempo devem ser removidas.

Uma vez que os computadores registram as suas ligações de endereços com o servidor ATMARP, o servidor mantém a entrada por no mínimo 20 minutos. Depois de 20 minutos, o servidor examina a entrada. Se nenhum circuito existe para o computador que gerou a entrada, o servidor deletará esta entrada da tabela. Se o computador que gerou a entrada manteve o circuito virtual aberto, o servidor revalida a entrada. O servidor envia uma requisição InATMARP e espera uma resposta. Se a resposta confirmar a informação da antiga entrada na tabela, o servidor "reseta" um *timer* e espera por mais 20 minutos. Se a resposta InATMARP não confirmar a informação da antiga entrada na tabela, o servidor fecha o circuito virtual antigo e remove a entrada na tabela.

Para reduzir o tráfego, o ATMARP permite utilizar somente um circuito virtual para todas as comunicações com o servidor ATMARP. Quando um *host* envia uma requisição ATMARP, a requisição contém os seus endereços IP e ATM. O servidor pode extrair esta informação e utilizá-la para revalidar a entrada na tabela. Assim, se um *host* envia mais de uma requisição ATMARP a cada 20 minutos, o servidor não necessitará enviar para o *host* uma requisição InATMARP.

Um host ou um roteador também usam temporizadores para invalidar informações obtidas de um servidor ATMARP. Em particular, as normas especificam que um computador pode manter uma ligação obtida de um servidor ATMARP por até 15 minutos. Quando os 15 minutos expirarem, a entrada deve ser removida ou revalidada. Se uma ligação de endereço expira e o *host* não tem um circuito virtual aberto para o destino, o *host* remove a entrada de seu *cache* ARP. Se um *host* tem um circuito virtual aberto para o destino, o *host* preocupa-se em revalidar a ligação de endereço.

A expiração de uma ligação de endereços pode atrasar o tráfego. Isto ocorre porque um host ou roteador deve parar de enviar dados para qualquer destino para o qual a ligação de endereços tenha expirado, até que a ligação tenha sido revalidada.

O método que um *host* usa para revalidar uma ligação depende do tipo de circuito virtual que está sendo utilizado. Se o *host* pode encontrar o destino sobre um PVC, o *host* envia uma requisição InATMARP pelo circuito virtual e espera uma resposta. Se o *host* tem um SVC aberto para o destino, o *host* envia uma requisição ATMARP para o servidor ATMARP.

2.1.5.7 Sinalização de Suporte para IP sobre ATM

A [RFC 1755] descreve a sinalização ATM necessária para suportar implementações do IP sobre ATM clássico (*Classical IP over ATM*), descritas na [RFC 1577]. Os pontos finais ATM (*ATM Endpoints*) irão incorporar serviços de sinalização ATM como especificados na [UNI 3.1] do Fórum ATM. As implementações de IP sobre ATM devem utilizar os serviços de entidades locais de sinalização ATM para estabelecer e finalizar conexões ATM.

Em um ambiente de conexão virtual comutada (S*witched Virtual Connection - SVC Environament),* as conexões de canal virtual ATM (*ATM Virtual Channel Connections - VCCs*) são dinamicamente estabelecidas quando necessário. Isto é conseguido graças a um protocolo de sinalização de controle de conexões ATM que opera entre sistemas finais ATM (*ATM Endsystems*) e a rede ATM. As entidades de sinalização usam o protocolo de sinalização para estabelecer e realizar chamadas (associação entre pontos finais ATM) e conexões (VCCs). O procedimento de sinalização inclui o uso de endereços para localizar pontos finais ATM e a alocação de recursos da rede para a conexão, bem como a indicação e a negociação entre pontos finais ATM visando a seleção dos protocolos e de seus parâmetros nos extremos da conexão. A [RFC 1755] descreve como o protocolo de sinalização é utilizado no suporte de IP sobre ATM e, em particular, a troca de informações feita pelo protocolo de sinalização para efetivar este suporte.

2.1.6 Considerações Finais

A operação do Modelo Clássico IP é muito simples, o que leva a muitas limitações. Uma destas limitações esta indicada pela palavra “clássico”. O que isto quer dizer é que o Modelo Clássico não possibilita que um pacote com destino externo a uma sub-rede IP seja enviado diretamente ao seu destino, sem precisar passar por um roteador *default.* Esta limitação gera ineficiência na rede, uma vez que o roteador torna-se um gargalo para a rede. Isto também impossibilita que uma conexão com um QoS pré-determinado seja estabelecida entre dois nós.

Como abordamos anteriormente, o grupo de trabalho *ROLC (Routing Over Large Clouds*) esta trabalhando no protocolo NHRP, que permite a conexão direta entre *hosts* de diferentes LISs através de um meio NBMA, como é o caso do ATM.

Outra limitação do Modelo Clássico é que ele não se preocupa com a questão da latência do estabelecimento de conexões virtuais, ao contrário do protocolo LAN *Emulation*, que tem um caminho *default* por onde os dados podem ser enviados até que a resolução de endereço, roteamento e estabelecimento de conexão sejam efetivadas.

O Modelo Clássico IP também não suporta *Multicast.* O protocolo “*Classical* IP *over* ATM” pode ser utilizado para resolver um endereço *multicast* IP para um endereço ATM. Porém não existe um mecanismo dentro de uma LIS para registro de um membro em um grupo *multicast*, ou como um endereço de grupo *multicast* pode ser mapeado para formar um *multicast* ATM.

Em síntese, o Modelo Clássico IP fornece uma solução bastante simples de interoperabilidade com redes ATM através do protocolo da camada de rede IP. Outras soluções mais elaboradas de interoperabilidade estão sendo investigadas pelo Fórum ATM e pela IETF. Entre elas o MPOA (*MultProtocol Over ATM*), que elimina muitas, senão a maioria das limitações do Modelo Clássico IP, permitindo o estabelecimento de conexões diretas através da rede ATM entre dois clientes de duas sub-redes (camada 3) diferentes.

2.2 Extensões ao Modelo Clássico do IETF

As principais restrições relativamente ao Modelo Clássico conforme proposto pelo IETF diz respeito à necessidade de rotear o pacote quando a origem e o destino da informação encontram-se em redes lógicas diferentes, e o fato de não suportar a comunicação *multicast*.
Na sequência serão apresentadas propostas no âmbito do IETF para contornar estas restrições.

2.2.1 IP Multicast Sobre ATM

No caso da comunicação *multicast* um *host* envia uma informação a qual, através de técnicas diversas, alcança todos os membros do grupo endereçados pela informação. Como a rede ATM não conhece o protocolo IP e o endereço Classe D utilizado para o *multicasting*, é necessária uma forma de mapeamento entre um endereço IP *multicast* e os endereços ATM dos membros do grupo *multicast* do IP.
O servidor MARS (*Multicast Address Resolution Server*) atua como uma central de registro dos membros dos grupos multicast e realiza a resolução do endereço IP *multicast* nos respectivos endereços dos membros do grupo.
Como no ATM a conexão ponto-multiponto unidirecional é a única forma de suportar transmissão *multicast*, a introdução dos participantes na conexão é importante para o gerenciamento do serviço *multicast*. A interface UNI 3.1 permite que somente a origem adicione folhas a uma conexão ponto-multiponto. Isto significa que a raiz deve ter conhecimento dos endereços ATM das folhas que participarão da conexão. Na interface UNI 4.0 é permitido que uma folha introduza-se unilateralmente em uma conexão ponto-multiponto.
A realização do *multicast* em redes ATM possui duas possibilidades. Na primeira, o *host* que deseja enviar a informação estabelece uma conexão ponto-multiponto com os outros *hosts* que fazem parte do grupo *multicast*. Cada membro do grupo que deseja transmitir dados *multicast* irá originar uma conexão ponto-multiponto para os membros do grupo conforme ilustrado na Figura 9. Esta solução denomina-se malha de VCs *(VC-Mesh*).
A segunda possibilidade caracteriza-se pelo emprego de um servidor denominado MCS (*Multicast Server*). Nesta solução um *host* que deseja transmitir uma informação *multicast* envia-a ao MCS através de uma conexão ponto a ponto, e este

retransmite-a através de uma conexão virtual ponto-multiponto a todos os membros do grupo *multicast* conforme ilustrado na Figura 10.
A diferença entre as duas soluções anteriores relaciona-se à quantidade de conexões que devem ser gerenciadas. No caso da solução *VC-Mesh* há uma grande quantidade de VCs com o consequente custo que isto representa em termos de *buffers*, blocos de controle nos comutadores e *overhead* de sinalização. Por outro lado, a solução VC-Mesh oferece melhores resultados do ponto de vista do desempenho.

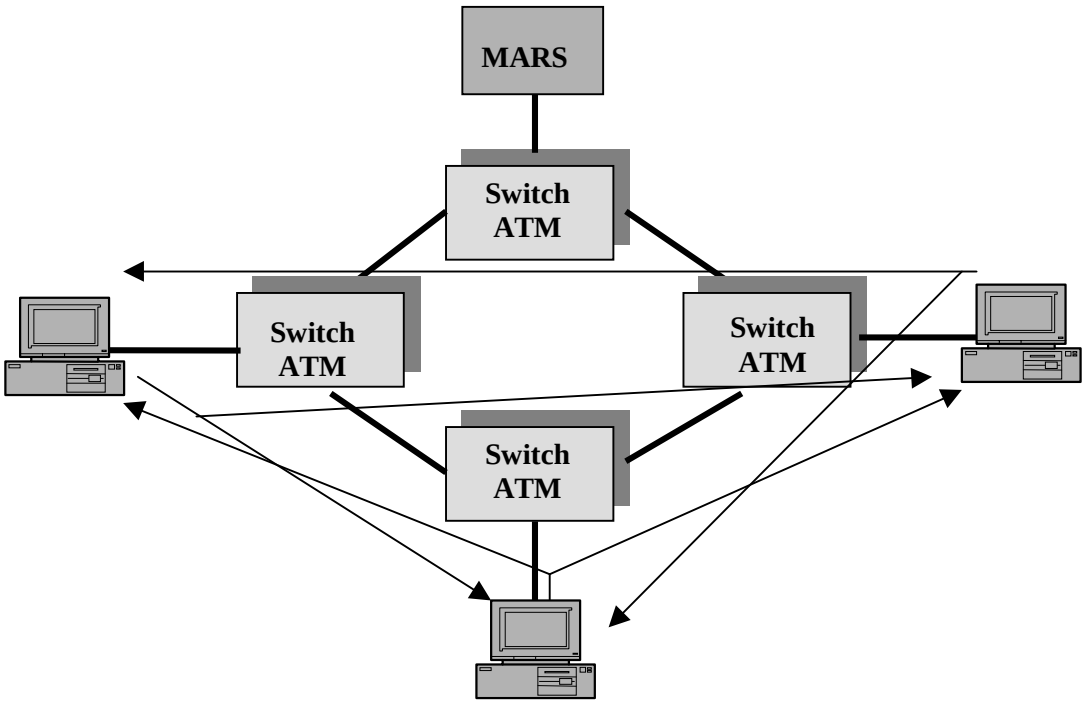


Figura 9: *VC-Mesh.*

No caso da solução através do servidor MCS existe um número reduzido de conexões virtuais para o grupo *multicast*. No caso de alguma alteração de membros do grupo isto afeta somente a conexão ponto-multiponto do servidor MCS, diferentemente da solução *VC-Mesh* onde a introdução/retirada de um membro afeta todas as conexões ponto-multiponto que estejam operando naquele momento. O maior problema da solução baseada no servidor MCS está relacionada ao desempenho, isto porque o tráfego *multicast* deve ser encaminhado através de uma conexão ponto a ponto ao servidor MCS, onde as células são reagrupadas em um quadro AAL5 antes da informação ser retransmitida ao grupo *multicast* pelo MCS.

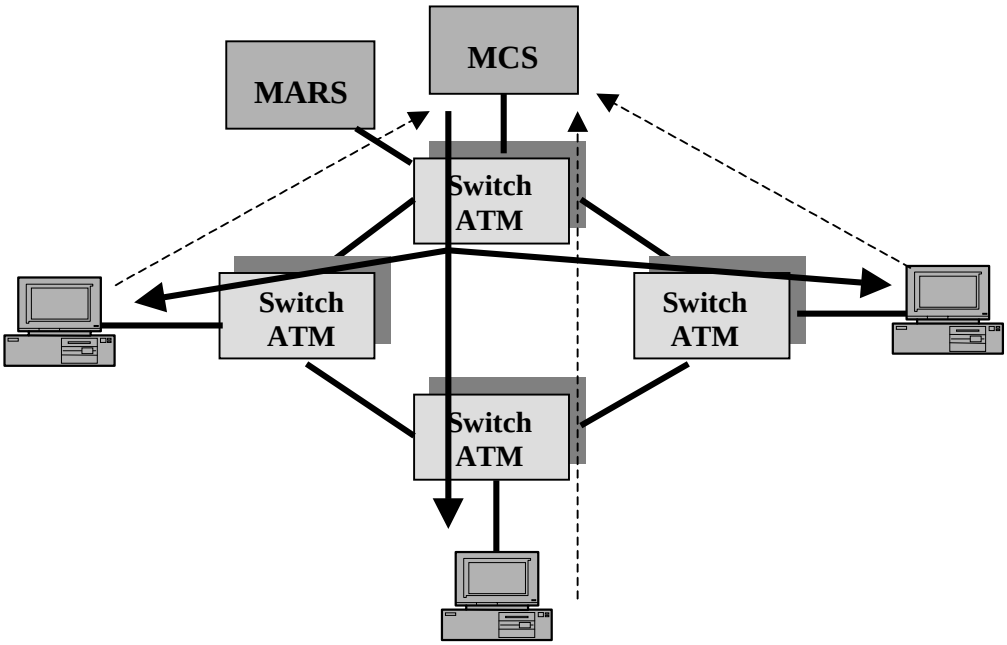


Figura 10: Servidor Multicast (MCS).

2.2.1.1 O Servidor MARS

Este servidor gerencia e dissemina as informações relativas aos membros do grupo *multicast*. Quando um *host* necessita enviar uma informação *multicast*, o servidor MARS é contactado para resolver o endereço *multicast* nos endereços ATM dos membros do grupo. Na realização desta função o servidor MARS introduz o conceito de *cluster* que corresponde ao conjunto de *hosts* ATM (ou roteadores) que compartilham aquele servidor MARS. Em geral um *cluster* corresponde a uma LIS. Existem trabalhos que procuram estender o conceito de *cluster* de modo a cobrir várias LISs, entretanto há necessidade de incluir o roteamento *multicast* o que é uma área ainda aberta a novas soluções.

O MARS pode ser visto como uma extensão ao servidor ATMARP onde, no lugar de manter uma tabela contendo pares de endereço ATM e endereço IP, ele gerencia uma tabela que contém endereços de grupo IP e os endereços ATM dos membros do *cluster* que fazem parte dos respectivos grupos.

A figura 12 ilustra o relacionamento dos *hosts* de um *cluster* com o MARS associado. Neste caso podemos observar a existência de conexões ponto a ponto entre cada membro do *cluster* e o MARS e uma conexão ponto multiponto entre o servidor MARS e os membros do *cluster*.

O cliente que deseja fazer parte de um grupo multicast estabelece uma conexão ponto a ponto com o MARS e envia uma mensagem MARS_JOIN com o endereço multicast 224.0.0.1, o qual é reservado pelo serviço multicast do protocolo IP para

controle da participação do *host* no protocolo *multicast* no qual(ais) ele está interessado. Sugere-se que esta conexão ponto a ponto seja encerrada caso não seja utilizada após um certo período de tempo.
A forma de operação do MARS pode tanto suportar o esquema baseado no uso do *VC-Mesh* ou do servidor MCS. A decisão de qual método utilizar é transparente ao *host*.

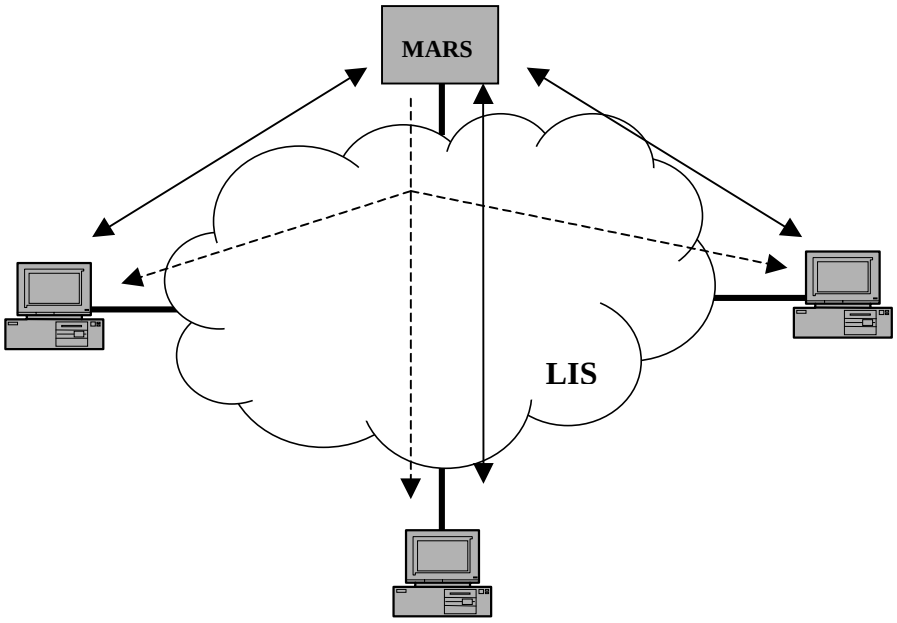


Figure 11: *Cluster* MARS.

Servidor Multicast (MCS)

Neste caso, o servidor que deseja servir um grupo multicast registra-se no MARS. Esta mensagem permite ao MARS construir um mapa de servidores *multicast* com a finalidade de facilitar a escalabilidade do serviço, ou seja, um endereço multicast pode ser suportado por vários servidores MCS. O MARS inclui o MCS que se registrou em um VC de controle ponto-multiponto denominado *ServerControlVC*. Esta conexão virtual é utilizada para informar ao(s) MCS(s) eventuais alterações que ocorram com os membros do grupo. Os endereços dos vários servidores MCS são retornados ao *host* que deseja transmitir no endereço *multicast* e que enviou a solicitação de resolução do endereço *multicast* ao MARS. Baseado neste mapa de endereços o *host* estabelece uma conexão ponto a ponto no caso de um único servidor MCS, ou uma conexão ponto-multiponto caso o MARS retorne os endereços de vários MCSs. O *host* transmite neste conexão os pacotes *multicast*. Os servidores MCS possuem VCs ponto-multiponto com membros pertecentes ao grupo *multicast* e através dos quais os pacotes são re-enviados aos membros do grupo. O MARS não toma parte do encaminhamento *multicast* dos dados.

Malha de VCs (VC-Mesh)

Quando o nó deseja fazer parte de um grupo *multicast* ele registra-se no MARS e este coloca-o como uma folha no VC de controle ponto-multiponto denominado *ClusterControlVC*. Esta conexão é utilizada para informar quando da alteração de membros do grupo.

O nó que deseja transmitir para um grupo *multicast* recebe do MARS, quando do envio de uma requisição de resolução do endereço *multicast*, uma resposta com uma lista de endereços ATM dos nós que se registraram no grupo. Desta forma o nó transmissor pode construir uma conexão ponto-multiponto e enviar o dado *multicast* na conexão. Desta maneira os pacotes são encaminhadas diretamente para cada membro do grupo. Da mesma forma como no caso do servidor *multicast*, o MARS não toma parte do encaminhamento dos pacotes.

2.2.2 Comunicação inter-subredes

Como discutimos anteriormente, no IP-Clássico sobre ATM o termo *clássico* evidencia o modelo original do protocolo IP no qual pacotes entre subredes diferentes devem passar necessariamente por um roteador. Neste caso a conectividade do ponto de vista do ATM, ou seja, a possibilidade de abrir uma conexão ATM fica restrita a membros de uma mesma LIS. Como a tendência é o crescimento da rede ATM, imagina-se a possibilidade de que várias LISs façam parte de uma mesma rede ATM. Seria interessante neste caso a possibilidade de que algumas aplicações possam abrir uma conexão ATM entre hosts que estejam em LISs diferentes em uma mesma rede NBMA. Isto permite evitar passar por roteador(es) para alcançar o destino, como no IP-Clássico, otimizando a comunicação para aplicações com restrições temporais como no caso do tráfego multimídia.

Com o objetivo de resolver esta restrição do modelo clássico, o IETF está discutindo o protocolo NHRP (*Next Hop Resolution Protocol*). Trata-se de um protocolo para resolução de endereços envolvendo *host* origem e *host* destino situados em redes lógicas (LISs) diferentes. Esta solução não significa abandonar o papel do roteador, o qual continua a realizar funções de encaminhamento e controle em muitos casos.

Além das questões próprias da comunicação IP internamente à subrede lógica como, endereçamento, resolução de endereço intra-subrede, encapsulamento de pacotes e multicast/broadcast, outras questões surgem no caso da comunicação IP entre subredes. Dentre estas questões podemos destacar: resolução do endereço remoto, protocolos de roteamento e a decisão entre rotear *hop-by-hop* ou estabelecer uma conexão SVC.

No modelo clássico a rede ATM é encarada como um protocolo de enlace qualquer, no caso, de alta velocidade, que é utilizado para permitir a conectividade entre *hosts* e roteadores em uma mesma LIS. Roteadores que fazem parte de uma mesma LIS estabelecem VCs entre si de modo a que o tráfego continue fluindo na forma *hop-by-hop* a partir da LIS de origem até à LIS de destino. Mesmo que todos os hosts e

roteadores estejam em uma mesma rede ATM (rede NBMA) o tráfego entre LISs continua a fluir através dos roteadores conforme ilustrado na Figure 12.
A dificuldade do modelo clássico deve-se ao fato de que ao fluir o tráfego pelo roteador é necessária a remontagem das células em pacotes correspondentes para a realização do roteamento. Outro aspecto negativo é que a conexão virtual envolvendo o(s) roteador(es) no percurso do pacote da LIS origem até a LIS do destino compromete a utilização da QoS para aquele pacote, ou seja, não é viável garantir uma QoS operando na forma sem conexão do modelo clássico.
A solução consiste na utilização da característica da orientação à conexão do ATM. Neste caso, um host em uma LIS poderia abrir uma nova conexão ATM com o host destino em outra LIS evitando desta forma que os pacotes, na forma de células, passem pelo roteador, ou seja, criando um atalho (*cut-through*) conforme ilustra a . Esta opção seria de grande interesse para as aplicações que necessitam dos aspectos de QoS suportados pelo ATM.

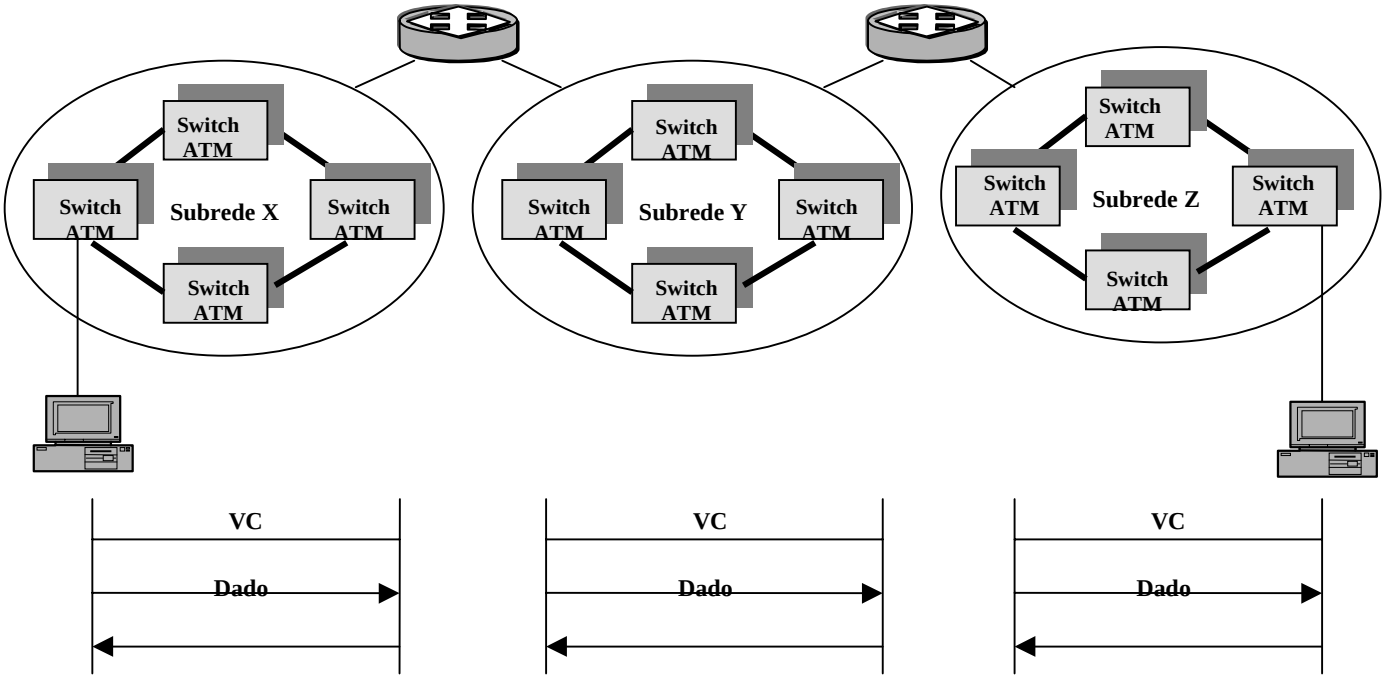


Figure 12: IP sem conexão sobre ATM.

Esta solução, entretanto, não deve ser utilizada como solução para todos os casos. Isto porque como a sua utilização implica na utilização da sinalização ATM e manutenção da informação do estado da conexão, mesmo para aquelas aplicações que possuem um tráfego limitado a poucos pacotes, não seria uma forma otimizada de gerenciar os recursos da rede.
A melhor maneira de otimizar o uso dos recursos da rede consiste na combinação das duas abordagens. Para as aplicações que requerem um serviço do tipo melhor esforço o tráfego é encaminhado via roteador baseado no modelo tradicional do IP.

Aplicações que demandam QoS devem procurar estabelecer um VC diretamente com o destino (*atalho*) contornando a passagem pelo(s) roteador(es).
No caso do estabelecimento de um atalho em LISs diferentes é necesário um procedimento para resolução de endereço remoto já que o esquema de resolução de endereço baseado no servidor ATMARP não é suficiente para este caso pois limita-se ao contexto de uma única LIS

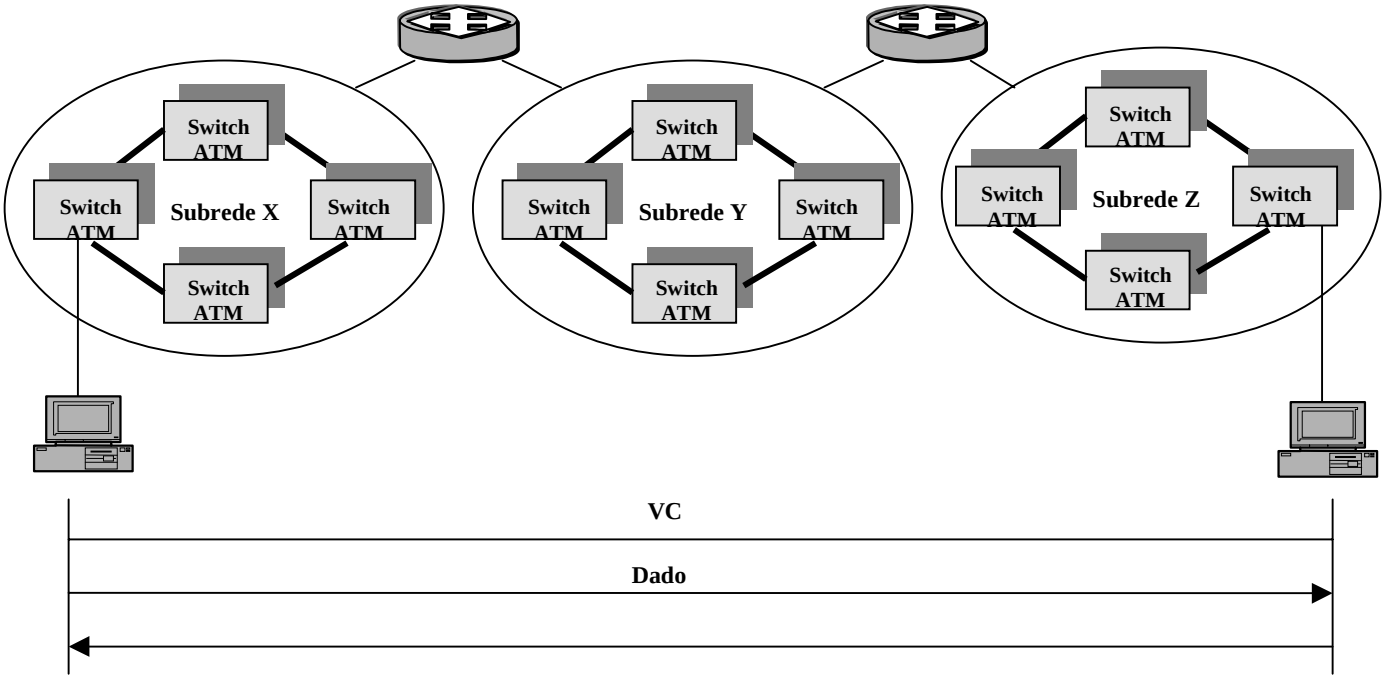


Figure 13: IP orientado a conexão sobre o ATM.

2.2.2.1 Protocolo NHRP (*Next Hop Resolution Protocol*)

O NHRP é um protocolo para resolução de endereço IP em endereço ATM em uma NBMA contando múltiplas redes lógicas (LISs). O NHRP foi proposto pelo grupo ROLC (*Routing Over Large Clouds*) do IETF como uma extensão ao IP-Clássico. Entretanto, pode ser utilizado para qualquer outro protocolo do nível 3. A sua utilização foi incluída na especificação MPOA (*Multiprotocolo Over ATM*) do ATM-Fórum e que será discutida posteriormente neste relatório. De forma geral, podemos dizer que o NHRP pode ser usado para qualquer protocolo de rede (IP, IPX, Appletalk, etc.) em qualquer rede do tipo NBMA (Frame-Relay, SMDS, X-25 e ATM).
O NHRP pode ser usado por uma estação origem, conectada a uma subrede NBMA, para a determinação do endereço da camada de rede e o endereço da subrede NBMA do "*NBMA Next Hop*" na direção da estação destino. Se o destinatário estiver conectado diretamente à subrede NBMA, então o "*NBMA Next Hop"* é a própria

estação destino. Caso contrário, o "*NBMA Next Hop"* é um roteador de saída da subrede NBMA que está mais próxima a estação de destino.
Uma subrede NBMA pode também ser entendida de outra forma: através do conceito de LAGs (*Logical Address Groups*). A diferença principal entre o modelo LAG e o LIS é que este último está baseado nos endereços unicamente para o encaminhamento dos pacotes, enquanto o LAG não está baseado nos endereços, mas na QoS e características de tráfego. Duas entidades dentro do LAG podem estabelecer uma conexão direta, independentemente dos seus endereços. Para que existam subredes como a descrita para o LAG, é necessário a existência de um mecanismo que faça a resolução dos endereços da camada de rede. Esta resolução deve ser feita independentemente dos endereços associados com as entidades. O NHRP implementa este mecanismo.

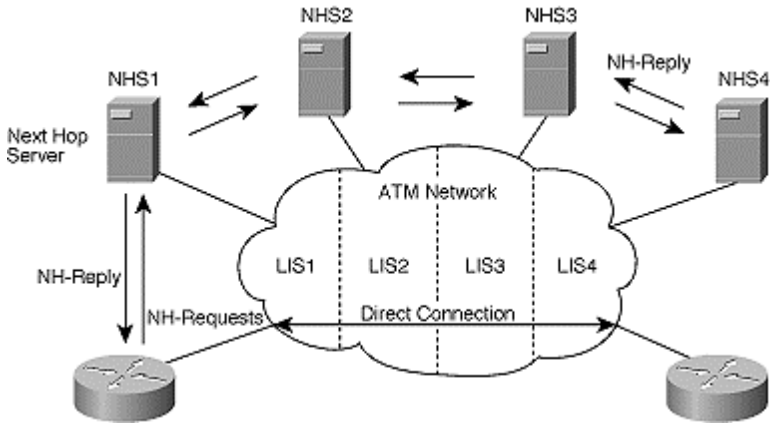


Figure 14: Protocolo NHRP.

O NHRP é formado por dois elementos básicos: servidor NHRP, denominado NHS (*Next Hop Server*), e clientes NHC (*Next Hop Clients*). O NHS oferece o serviço NHRP a clientes NHRP em uma rede NBMA. O NHS mantém em uma tabela *cache* os endereços IP e ATM dos dispositivos conectados à rede ATM. Na eventualidade do endereço IP do destino não fazer parte da NBMA, o NHS retornará o endereço de um roteador que permita encaminhar o pacote na direção do destino. Os clientes NHC acessam o NHS que os serve com o objetivo de obter os serviços NHRP e, em última instância, obter o endereço ATM que corresponde ao endereço IP do destino, ou corresponde a um roteador quando o destino encontra-se em outra NBMA ou , ainda, no caso de realizar alguma proteção caso as estações estejam na mesma NBMA mas em domínios administrativos diferentes.

Arquitetura do Serviço NHRP

Os clientes NHC em uma rede ATM obtêm de alguma forma o endereço do servidor NHS correspondente através, por exemplo, do acesso a um servidor de configuração. Este servidor pode ser utilizado para oferecer também os endereços de

outros servidores como o ATMARP e MARS. Os clientes executam em *hosts* ou roteadores conectados à rede ATM, onde o NHS do cliente será representado pelo seu roteador *default* ou seu roteador par. Ela também pode pertencer a múltiplas subredes NBMA, sendo então configurada com vários NHS.

Na fase de iniciação, os servidores NHRP (NHS) são configurados com os seus endereços IP e ATM, um conjunto de prefixos IP das redes atendidas pelo servidor e um identificador para a rede NBMA correspondente.

Os clientes registram-se no seu servidor NHRP através do envio de pacotes de registro que contêm os valores dos seus endereços ATM e IP.

Resolução de endereço através do NHRP

Um aspecto que deve ser destacado no caso do NHRP é que este não se trata de um protocolo de roteamento. O NHRP não troca informações de alcançabilidade como fazem os protocolos de roteamento tradicionais. O que provoca alguma confusão é o fato de que o NHRP necessita ser executado em cooperação com os protocolos de roteamento.

Para que um endereço seja resolvido, uma sequência de operações deve ser realizada. Quando uma estação origem precisa enviar um pacote para uma estação destino que ela não possui o endereço, o protocolo NHRP é disparado. A estação origem determina o próximo salto para a estação destino através dos processos normais de roteamento (usando, por exemplo, a rota *default*, que neste caso representa o *next hop* do endereço da camada de rede destino). Se o próximo salto pode ser encontrado através da sua interface com a subrede NBMA, então a estação origem constrói uma requisição NHRP contendo o endereço da camada de rede da estação destino, o seu endereço de camada de rede como origem e a informação de endereçamento de sua NBMA. A estação origem pode requerer que a resposta seja *autorizada*.Uma resposta autorizada, dentro do NHRP, é aquela resposta que provém diretamente do NHS (Next Hop Server) que serve a estação destino; as demais respostas, vindas de NHS intermediários (que possuem informações sobre a requisição armazenada em *cache*), são consideradas não-autorizadas ou "*non-authoritative answer*". Um cliente NHS ao enviar uma requisição de resolução de endereço ao seu servidor NHS deverá especificar se ele deseja receber uma resposta autorizada ou uma resposta não-autorizada.

A requisição NHRP é enviada pela estação origem em direção ao destino, usando o endereço NBMA do roteador do próximo salto. Se a resolução do endereço para o envio do pacote originou-se da necessidade de transmitir dados, a estação origem pode querer enviar os dados ao longo do caminho que está sendo roteado em direção à estação destino. Quando o NHS recebe esta requisição de resolução de endereço, ele verifica se ele mesmo não serve a estação destino, ou seja, ele verifica se não existe uma entrada da estação destino em sua *cache*. Caso não exista, ele encaminha a requisição para outro NHS. Se este último NHS serve a estação destino, ele resolve o endereço da requisição e gera uma resposta positiva (esta resposta no caso é a resposta autorizada). Esta resposta contém o endereço da camada de rede e o endereço NBMA da estação destino, sendo então enviado de volta para a estação origem. Se a estação destino não estiver na mesma subrede

NBMA, o endereço da camada de rede colocado na resposta será o do roteador de saída, através do qual os pacotes para a estação destino são transmitidos.

Quando o primeiro NHS recebe de volta a resposta, ele pode armazenar em *cache* a informação para que posteriores requisições sejam respondidas através de respostas não-autorizadas.

A resposta NHRP pode também ser feita diretamente para a estação origem, não passando pelos NHSs que transmitiram a requisição. Isto é feito para reduzir o tempo de resposta, porém a informação não poderá ser armazenada pelos NHSs intermediários. O processo de envio de uma requisição NHRP é repetido até ser satisfeita ou então, até a ocorrência de um erro.

Uma requisição e uma resposta NHRP não podem cruzar a fronteira da subrede lógica NBMA. Um identificador da subrede NBMA precisa ser incluído como extensão da requisição. Então, o tráfego da camada de rede para fora da subrede lógica NBMA sempre precisa passar por um roteador na fronteira.

Um campo adicional, chamado de *Route Record*, é colocado na requisição NHRP com a finalidade de filtragem da subrede, capacidade de diagnóstico e, principalmente, para a detecção de *loops*. Neste campo são colocados os endereços de todos os NHSs intermediários pelos quais uma requisição NHRP passou entre a fonte e o destino.

Os NHSs são capazes de tomar conhecimento da existência de outros NHSs através da sua participação em trocas de informações de protocolos de roteamento intra-domínio e inter-domínio. Também podem ter este conhecimento através da configuração estática dos endereços de outros NHSs.

Principais mensagens

Um pacote NHRP é constuído de três partes: uma parte fixa, uma parte obrigatória e uma parte de extensão. A parte fixa é comum em todos os tipos de pacote NHRP, e contém aqueles elementos de tamanho fixo que devem estar sempre presentes. A parte obrigatória precisa estar presente em todos os pacotes, porém varia de acordo com o tipo de pacote. A última parte também varia de acordo com o tipo de pacote, mas não necessita estar presente.

Os pacotes NHRP são encapsulados de acordo com o formato usado na NBMA sobre a qual o pacote é transmitido. Em particular, o ATM usa o encapsulamento LLC/SNAP ou não usa encapsulamento se for um VC (Virtual Channel) dedicado.
A parte obrigatória contém informações específicas de operação e variam de tamanho, de acordo com o tipo de pacote. Seus tipos são apresentados na tabela 2.

2.2.2.2 Perspectivas do NHRP

O protocolo NHRP é voltado para a entrega de pacotes IP tendo como destino um endereço *unicast*. Neste sentido há necessidade do protocolo ser estendido de modo a incorporar suporte à operação *multicast*/*broadcast*. Outro item importante

é o fato do NHRP ser muito específico relativamente ao IP, por exemplo, as mensagens NHRP são enviadas como dados em pacotes IP.

Um aspecto que tem merecido atenção é relacionado à escalabilidade do protocolo. Esta escalabilidade deve ser enfocada em 3 níveis distintos:

1. Nível do cliente;
2. Nível da LIS;
3. Nível do domínio.

Nome do Pacote	*Descrição da Função do Pacote*
NHRP Next Hop Resolution Request	É utilizado na requisição de resolução de um endereço. É enviado por uma estação para o NHS.
NHRP Next Hop Resolution Replay	É a resposta enviada por um NHS, ao longo do caminho para o destino, para a estação solicitante sobre a informação requerida.
NHRP Registration Request	É enviado de uma estação para um NHS para notificá-lo sobre a informação NBMA da estação. É utilizado para registar o protocolo e o endereço NBMA da estação.
NHRP Registration Replay	É enviado pelo NHS ao cliente em resposta ao envio do **NHRP** Registration Request.
NHRP Purge Request	É enviado para invalidar uma informação armazenada em *cache* numa estação. Este pacote é enviado de um NHS para uma estação para apagar uma informação previamante armazenada. Também pode ser enviada a partir de uma estação para o NHS, onde se registrou, invalidando seu registro.
NHRP Purge Replay	É enviado para assegurar ao emissário da mensagem **NHRP** Purge Request que toda informação armazenada de um determinado tipo foi removida.
NHRP Error Indication	É usado para transmitir uma indicação de erro ao emissor de um pacote **NHRP**.

Tabela 2: Principais mensagens NHRP.

Do ponto de vista do cliente a escalabilidade do NHRP é limitada pela capacidade de processamento e da memória da placa que conecta o cliente à rede NBMA. Por exemplo, quando o cliente situa-se em uma interface NBMA do roteador, estas podem constituir uma restrição importante. Isto deve-se ao fato de que os roteadores têm de agregar tráfego originado em múltiplos pontos o que cria um número grande de SVCs na interface NBMA do roteador.

No nível da LIS, a principal questão está relacionada à manutenção e manipulação de uma grande quantidade de informação associada ao mapeamento do endereço de rede no respectivo endereço NBMA no caso de grandes LISs. Nesta situação o NHRP pode utilizar vários NHSs em uma mesma LIS, o que traz como consequência a necessidade de um protocolo para sincronização da tabela *cache* dos vários NHSs na mesma LIS.

No nível do domínio NHRP a escalabilidade do protocolo é intimamente associada à escalabilidade do protocolo de roteamento usado pelo NHRP. No caso dos protocolos de roteamento dinâmico esta escalabilidade é boa.

Consequentemente, quando o NHRP trabalha associado a este tipo de protocolo a sua escalabilidade também é boa. Isto sob a hipótese de que todos os roteadores ao longo do caminho implementam o NHRP. Caso um roteador não implemente o protocolo, uma requisição NHRP neste roteador será descartada sem qualquer informação adicional. Neste caso o atalho não poderá ser estabelecido e a solução será utilizar o encaminhamento da informação *hop-by-hop*.

Caso utiliza-se um roteamento estático, não será necessário que todos os roteadores implementem o NHRP. Neste caso, os roteadores que devem processar as mensagens de controle são especificados pelo roteamento estático, e aqueles que não são incluidos neste caminho não precisam executar o protocolo. Obviamente a solução estática não oferece uma escalabilidade adequada.

O NHRP na forma como é definido atualmente não é suficiente para suprimir a ocorrência de *loops* quando usado na comunicação roteador-roteador. Trabalhos são desenvolvidos atualmente com o objetivo de eliminar este problema.

2.3 Modelo Lan Emulation (ATM-Fórum)[4]

O objetivo essencial do serviço de *LAN Emulation* (**LANE**) na forma proposta pelo ATM-Fórum consiste na introdução do ATM preservando todo o investimento já realizado no caso das redes locais. O requisito básico é que os protocolos atuais que executam no nível de rede não tenham que sofrer qualquer alteração caso executem em uma rede Ethernet ou Token-Ring, ou caso executem sobre uma infraestrutura caracterizada por uma nuvem ATM.

Vários modelos são possíveis para a realização da emulação de LAN. No caso mais simples pode-se utilizar um servidor com uma conexão ponto-multiponto para todos os membros da LAN emulada. Quando um membro da LAN deseja enviar uma mensagem, ele a envia para o servidor que, em resposta, envia a mensagem para todos os membros da LANE. Este modelo impõe um *overhead* substancial no sistema por não utilizar a característica do ATM de levar o tráfego diretamente da máquina de envio para a máquina destino através da nuvem de chaves ATM.

A especificação atual da *LAN Emulation* proposta pelo ATM Fórum fornece uma solução genérica que permite a realização das funções de ponte e roteamento (*bridging* and *routing*), e tenta conservar a maioria do tráfego da rede em conexões ponto-a-ponto entre a máquina de envio e a máquina receptora.

A dificuldade maior em oferecer um serviço de emulação de LAN sobre uma infraestrutura ATM consiste no fato de que as redes locais tradicionais e a tecnologia ATM possuem diferenças marcantes. Desta forma, há a necessidade de introduzir uma funcionalidade acima da camada de adaptação ATM que crie para os protocolos do nível de rede uma interface compatível com as interfaces atualmente suportadas pelas redes locais (ODI, NDIS)[5].

Uma diferença essencial é que as LANs são não orientadas à conexão, enquanto as redes ATM suportam serviços orientados à conexão. Deste modo, a *mais importante função de um serviço de emulação de LAN é a provisão de um serviço não orientado à conexão.*

Além dos endereços MAC *unicast* identificando destinações únicas, o serviço de emulação de LAN tem que suportar o uso de *broadcast* e endereços MAC de grupo (*multicast*), porque *broadcast* e *multicast* são características nativas de meios compartilhados em LANs. Entretanto, o fornecimento de tal serviço sobre uma rede baseada em chaves com enlaces ponto-a-ponto como a rede ATM não é trivial.

Outra diferença importante entre as LANs existentes e as redes ATM é o tamanho das Unidades de Dados de Protocolo (Protocol Data Units - PDUs) usados para troca de informações. Enquanto as LANs usam quadros (ou mensagens) com tamanhos variáveis, o ATM é uma tecnologia baseada em quadros com tamanho fixo de 53 bytes (células) e, conseqüentemente, um quadro LAN usualmente não se encaixa numa célula ATM e tem que ser segmentado de forma semelhante à segmenteção do pacote em células no caso do IP-Clássico.

[4] Baseado na referência [Oli,96].

[5] ODI = *Open Data Link Interface;*
NDIS = *Network Driver Interface Specification.*

O ATM Fórum completou em 1995 a versão 1 da especificação LANE, onde o protocolo define mecanismos para emulação da IEEE 802.3/Ethernet e da IEEE 802.5/Token Ring.

O protocolo LANE define uma interface de serviço para os protocolos das camadas mais altas (isto é, camada de rede) idêntica às das LANs já existentes. Os dados enviados através da rede ATM são encapsulados em um formato de pacote LAN MAC apropriado. Em outras palavras, os protocolos LANE fazem uma rede ATM parecer e agir como uma LAN Ethernet ou Token Ring com uma taxa de transmissão muito mais elevada. O objetivo de realizar a emulação de LAN significa oferecer uma interface compatível com as interfaces atuais das redes locais e não emular os procedimentos de acesso ao meio como o CSMA/CD ou a passagem de token.

Como os produtos e serviços na linha do ATM estão se tornando largamente disponíveis, os profissionais da área estão procurando maneiras de obter os benefícios proporcionados pela alta velocidade e alta capacidade ATM, enquanto preservam elementos da infra-estrutura de rede existente. A especificação LANE do ATM Fórum define mecanismos que permitem às redes ATM coexistirem com sistemas mais antigos fornecendo um caminho de migração escalonável para ATM. O conceito de uma "Camada de Emulação de Rede Local" (*LAN Emulation Layer*) é introduzido em sistemas terminais que interfaceiam entre a LAN tradicional e o novo meio ATM. Isto permite ao usuário continuar a utilizar o *hardware* disponível enquanto migra para a nova arquitetura.

A interface LUNI (*LAN Emulation User-to-Network Interface*) permite a interoperabilidade entre os vários equipamentos que compõem a LAN emulada, basicamente, os clientes e os servidores LANE. A versão 1.0 da LANE define o modo padrão para um cliente resolver endereços MAC em endereços ATM e comunicar-se com outros clientes para envio de dados através da rede ATM. A versão especifica como o cliente LAN Emulation Client (**LEC**) interage com o LAN Emulation Server (**LES**) através da User-to-Network Interface (UNI).

A versão 2.0 do LANE está prevista para 1997 e definirá protocolos entre servidores para permitir que servidores de diferentes fabricantes interoperem aumentando a escalonabilidade e robustez da LANE, além de tentar contornar algumas restrições presentes na versão 1.0.

2.3.1 Visão Geral da Arquitetura

Uma LANE suporta a comunicação de quadros de dados de usuário similar a uma LAN física convencional. Uma ou mais LANs emuladas podem estar na mesma rede ATM. Entretanto, cada uma das LANs emuladas (**ELAN**-*Emulated Lan*) são independentes umas das outras e as estações não podem se comunicar diretamente através dos limites das LANs emuladas. A comunicação entre LANs emuladas é possível somente através de roteadores (Figura 15).

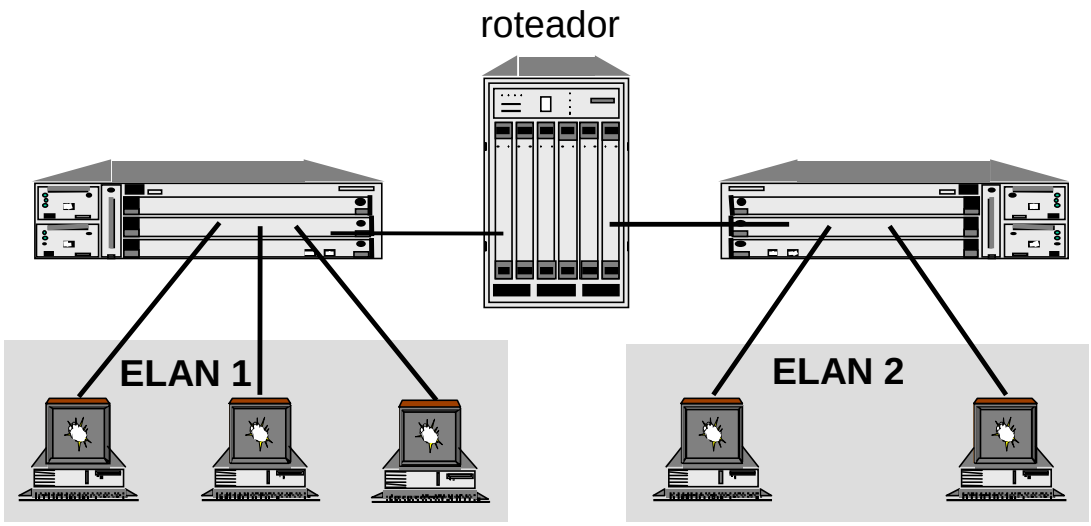


Figura 15: Interconexão entre ELANs diferentes

Uma LAN emulada pode ser do tipo Ethernet/IEEE 802.3 ou IEEE 802.5 (Token Ring). No cenário 1, Figura 16, a rede ATM interconecta-se a múltiplos segmentos Ethernet e *hosts* ATM (*ATM attached end-systems*). Neste cenário são possíveis as seguintes interconexões: *host* ATM para *host* ATM, *host* Ethernet para *host* ATM e *host* Ethernet para *host* Ethernet. No cenário 2, Figura 17, uma rede ATM interconecta-se a múltiplas redes Token Ring com as seguintes opções: *host* ATM para *host* ATM, *host* Token Ring para *host* ATM e *host* Token Ring para *host* Token Ring.

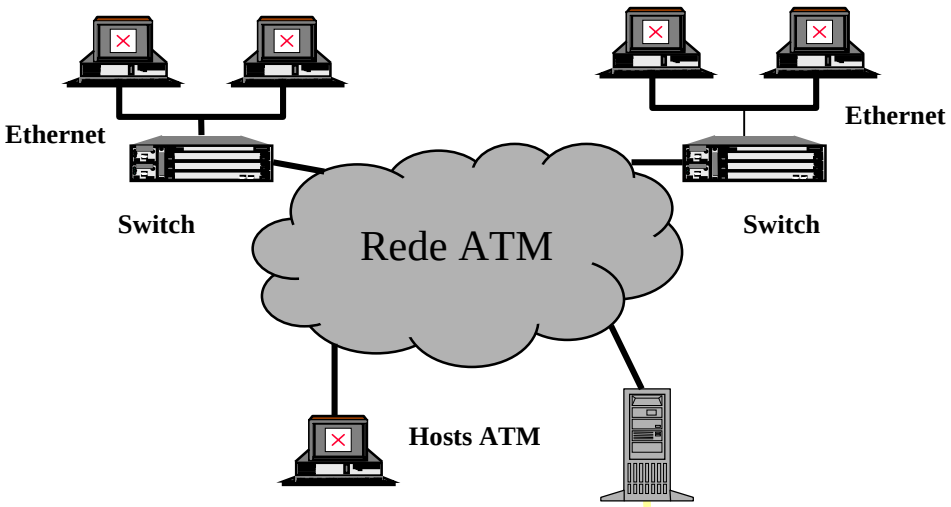


Figura 16: Cenário 1 - Ethernet e *hosts* ATM.

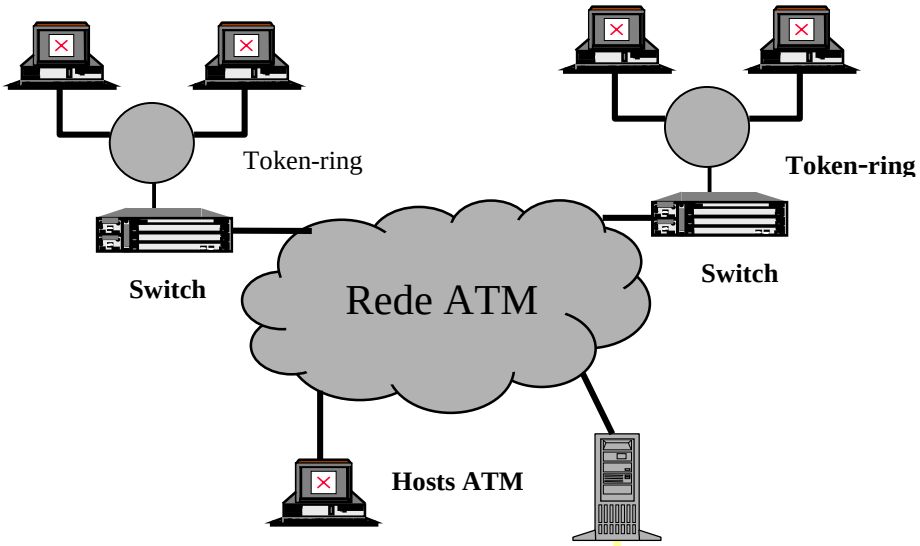


Figura 17: Cenário 2 - Token Ring e *Hosts* ATM**.**

Cada LAN emulada é composta por um conjunto de Clientes LANE (LECs-*Lan Emulation Client*) e um serviço de emulação de LANE. Este serviço consiste de um Servidor de Configuração LE (**LECS**-*LE Configuration Server*), um servidor de LAN emulada (***LES***-*LAN Emulation Server*) e um servidor de broadcast (***BUS***-*Broadcast and Unknown Server*). Cada LEC é parte de um dispositivo final conectado à rede ATM (*hosts*, roteadores, pontes e LAN *chaves*) e representa um conjunto de usuários identificados pelos seus endereços MAC. O serviço LE pode ser parte de uma estação ou uma chave ATM, podendo ser centralizado ou distribuído sobre um número de estações.

A participação em uma LAN emulada não é baseada na localização física do cliente mas sim na associação a um conjunto específico de serviços. Esta característica torna a LAN emulada adequada à construção e gerenciamento de LANs Virtuais (**VLANs**-*Virtual LANs*).

A comunicação entre LECs e entre um LEC e o serviço de LAN emulada é realizado sobre VCCs de controle e dados. As LANs emuladas operam em qualquer dos seguintes meios:

- Circuito Virtual Comutado (*SVC- Switched Virtual Circuit*);
- Circuito Virtual Permanente (*PVC- Permanent Virtual Circuit*);
- Mistura de SVC/PVC.

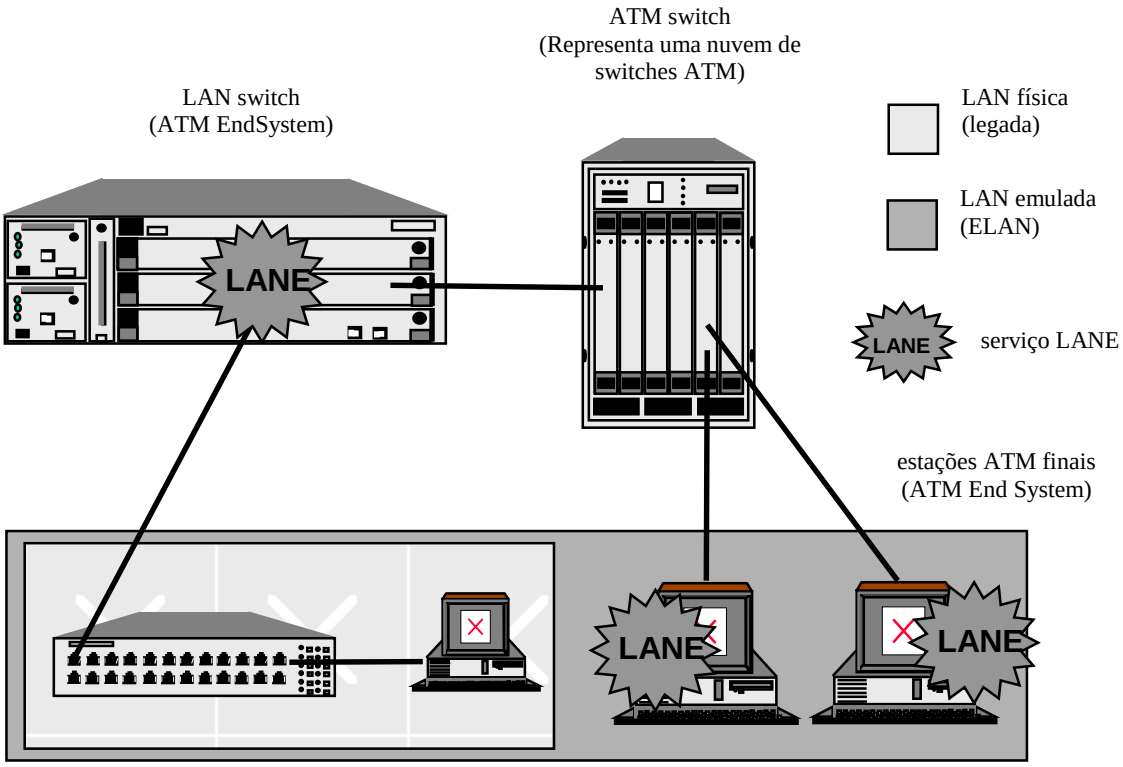


Figura 18: Arquitetura LAN Emulation

No caso de PVC, a camada de gerência é responsável tanto pelo estabelecimento como pelo encerramento das conexões, e tem completa responsabilidade de assegurar que a LAN emulada funcionará corretamente.

Na arquitetura proposta para a LANE as camadas interagem através de uma interface de serviço bem definida, fornecendo serviços que serão especificados a seguir (Figura 19). Os requisitos da interface são os seguintes:

1. A interface entre a camada LANE e a camada superior inclui facilidades para transmissão e recepção de quadros de dados dos usuários;
2. A interface entre a camada LANE e a camada de adaptação ATM (AAL) inclui facilidades para transmissão e recepção de quadros AAL5. As interfaces de ponto de acesso a serviços são identificadas por SAP-IDs (que possuem mapeamento um a um para VCCs);
3. A interface entre a entidade LANE e a entidade de gerenciamento de conexão inclui facilidades para requisitar a instalação e a liberação das conexões virtuais. Essa entidade manipula os SVCs e/ou PVCs.

A interface entre a entidade LANE e a camada de gerência inclui facilidades para iniciar e controlar a entidade LANE e o retorno de informação do status.

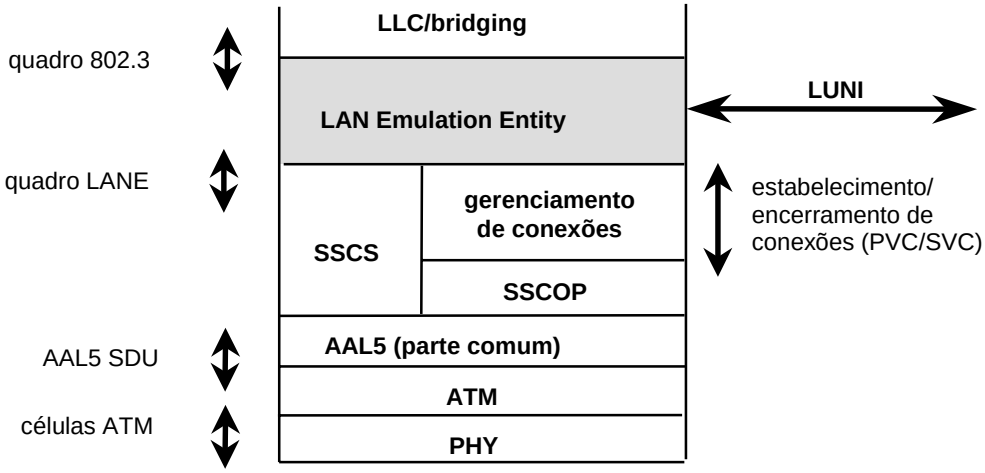


Figura 19: Arquitetura em camadas da LAN Emulation.

A Figura 20 ilustra um cenário onde um LEC, em um *host* ATM, interage com um LEC, em um *switch* ATM, o qual interconecta à rede ATM um *host* de rede local tradicional.

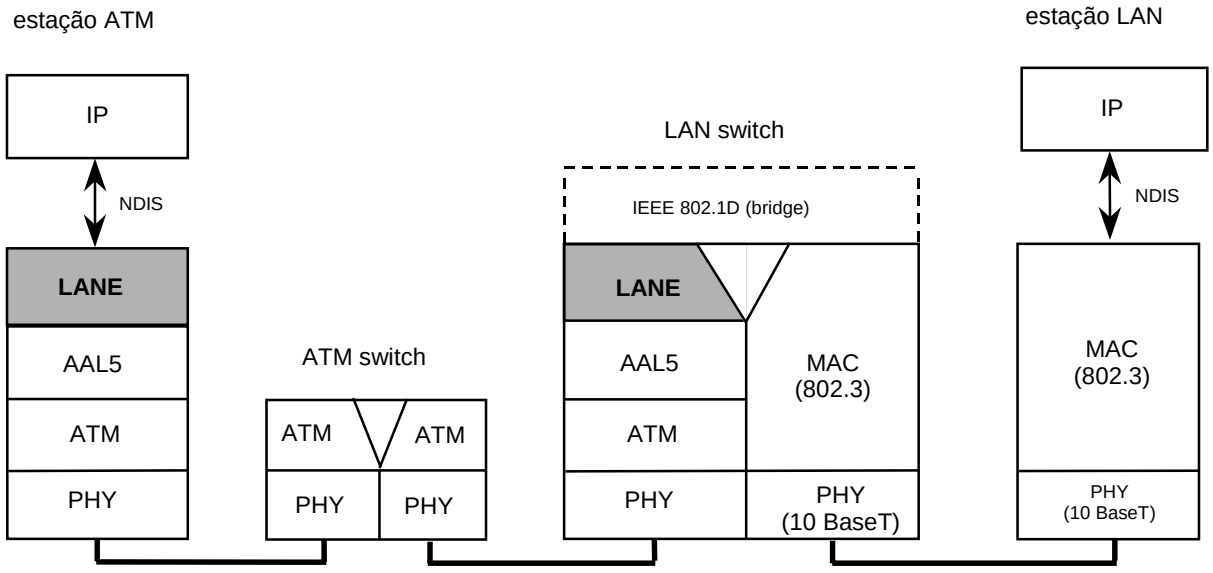


Figura 20: Arquitetura do protocolo LANE.

2.3.2 LAN Emulation User to Network Interface (LUNI):

Neste modelo de arquitetura, o LEC e o LES interagem através da interface usando PDUs e implementando as funções especificadas a seguir (Figura 21).

- Iniciação: obtenção do endereço ATM dos servidores que compõem o Serviço de LAN Emulada;
- Registro: informar ao serviço LANE os endereços MAC dos clientes ou os descritores de rota na origem (*Source-route bridge*);
- Resolução de endereço: obtenção do endereço ATM a partir do endereço MAC (*unicast* ou *broadcast*) do cliente destino;
- Transferência de dado.

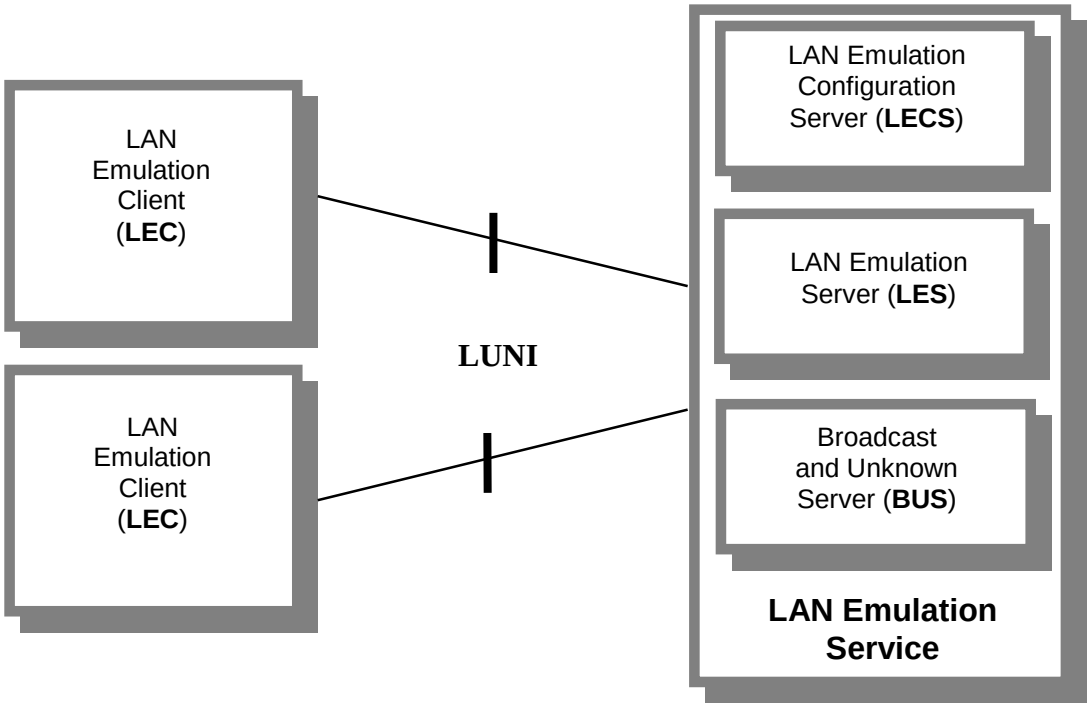


Figura 21: Interface Usuário-Rede (LUNI) da LAN Emulation.

2.3.3 Perspectiva de Implementação:

Usuários conectam-se ao serviço LANE via LECs. Os LECs são tipicamente implementados como parte do *driver* (software entre o Sistema Operacional e o hardware ATM) que controla a placa ATM.

A LAN emulada deve ser usada nas seguintes configurações:

1. Sistemas Intermediários (isto é, pontes, chaves de LAN ou roteadores). Estes equipamentos habilitam a comunicação entre LANs já existentes sobre redes *backbone* ATM;
2. *Hosts* ATM (isto é, estações de trabalho ou PCs). Estão habilitados a se comunicarem ou com estações conectadas às LANs tradicionais e interligadas na nuvem ATM através de um dispositivo intermediário.

O Serviço LANE deve ser implementado em um sistema intermediário ATM (roteador) ou em um *host* ATM. Alternativamente, ele pode ser "parte da rede ATM", isto é, implementado em comutadores (*chaves)* ou outros equipamentos específicos ATM.

2.3.4 Componentes da LANE

O serviço de LAN Emulation segue o modelo cliente/servidor, no qual múltiplos clientes LANE (LECs) utilizam o serviço LANE, este último suportado por três tipos de componentes: os servidores LES, BUS e LECS.

O protocolo LANE define a operação de uma única LAN emulada (ELAN). Múltiplas LANs emuladas (ELANs) podem coexistir simultaneamente em uma rede ATM, desde que as conexões ATM não colidam. Uma única LAN emulada emula o padrão Ethernet ou Token Ring, e apresenta os componentes descritos a seguir.

- **LAN Emulation Client (LEC)**

Um LEC executa a resolução de endereços, o envio de dados e outras funções de controle para um sistema final em uma única ELAN. Um LEC também provê uma interface de serviço de LAN padrão para as entidades de camada mais alta que interfaceiam com o LEC. Um sistema final que se encontra conectado a várias ELANs deverá apresentar um LEC por ELAN.

Cada LEC é identificado por um endereço ATM único e é associado a um ou mais endereços MAC através do endereço ATM. Caso o LEC seja uma NIC (*Network Interface Card*), ele terá um único endereço MAC correspondendo a um único endereço ATM. Porém, se o cliente estiver implementado em um LAN *switch*, ele apresentará vários endereços MAC referentes às portas do *switch* que fazem parte da LAN emulada correspodendo a um único endereço ATM.

Os LECs se comunicam com as funções do serviço de LAN Emulation através de dois tipos de *conexões de canal virtual* distintos (VCCs):

- conexões de controle: contendo mensagens administrativas, tais como requisições para configuração inicial e para obtenção de endereços de outros LECs;
- conexões de dados: manipulam todas as outras comunicações. Em particular, elas ligam clientes através de comunicação direta de dados unicast, e ligam clientes ao BUS para mensagens *broadcast* e *multicast*.

Os VCCs podem operar sobre circuitos virtuais chaveados alocados dinamicamente (SVCs), circuitos virtuais permanentes (PVCs), ou uma mistura de ambos.

Cada componente de rede pode suportar múltiplas instâncias de um LEC, permitindo que várias LANs emuladas existam simultaneamente em uma mesma rede física. Por exemplo, um roteador ATM encarregado do gerenciamento do tráfego entre duas LANs emuladas distintas deve suportar duas instâncias de um LEC, uma para cada LAN emulada.

É importante ressaltar que, se uma especificação de LANE inclui dois tipos de LAN emulada, uma Token Ring e uma Ethernet, isso não significa que seja permitida a conectividade direta entre o LEC que implementa a rede emulada Ethernet e um que implemente a rede emulada Token Ring.

- **LAN Emulation Server (LES)**

Os servidores da LAN emulada implementam as funções de controle para uma determinada ELAN através do registro e resolução de endereços para os LECs conectados à LANE. Há somente um LES por ELAN, sendo identificado por um endereço ATM único. O LES executa funções de controle, incluindo resolução de endereços de acesso ao meio (MAC) em endereços ATM. Os clientes devem registrar junto ao LES o seu endereco MAS e as LANs destino que eles representam. Um cliente também deve consultar ao LES quando necessitar resolver um endereço MAC e/ou um descritor de rota em um endereço ATM. O LES irá responder diretamente ao cliente ou enviar a requisição a outros clientes para que estes respondam.

- **Broadcast and Unknown Server (BUS):**

As redes locais são baseadas em envio *broadcast*. Dessa forma, a LAN emulada deve oferecer esse tipo de serviço. A função do BUS é enviar mensagens *broadcast* tais como mensagens de resolução de endereços IP. Cada LEC é associado a um único BUS por ELAN. Um BUS ao qual um LEC se conecta é identificado por um endereço ATM único. No LES, o endereço do BUS corresponde ao endereço MAC de *broadcast*, e esse mapeamento é geralmente configurado no LES. O servidor manipula os quadros de *broadcast* e *multicast*, bem como quadros para os quais o endereço MAC ainda não foi resolvido para endereço ATM.

Todos os LECs mantêm uma conexão ponto-ponto com o BUS da sua LANE e são folhas em uma conexão ponto-multiponto para a qual o BUS é a raiz. Isto permite aos clientes enviarem quadros antes que seja resolvido o endereço do destino e o consequente estabelecimento de uma conexão direta para o envio dos quadros. Esta característica mantém a presença de um serviço sem conexão de transmissão de dados típico das redes locais tradicionais. A Figura 22 ilustra a operação do BUS.

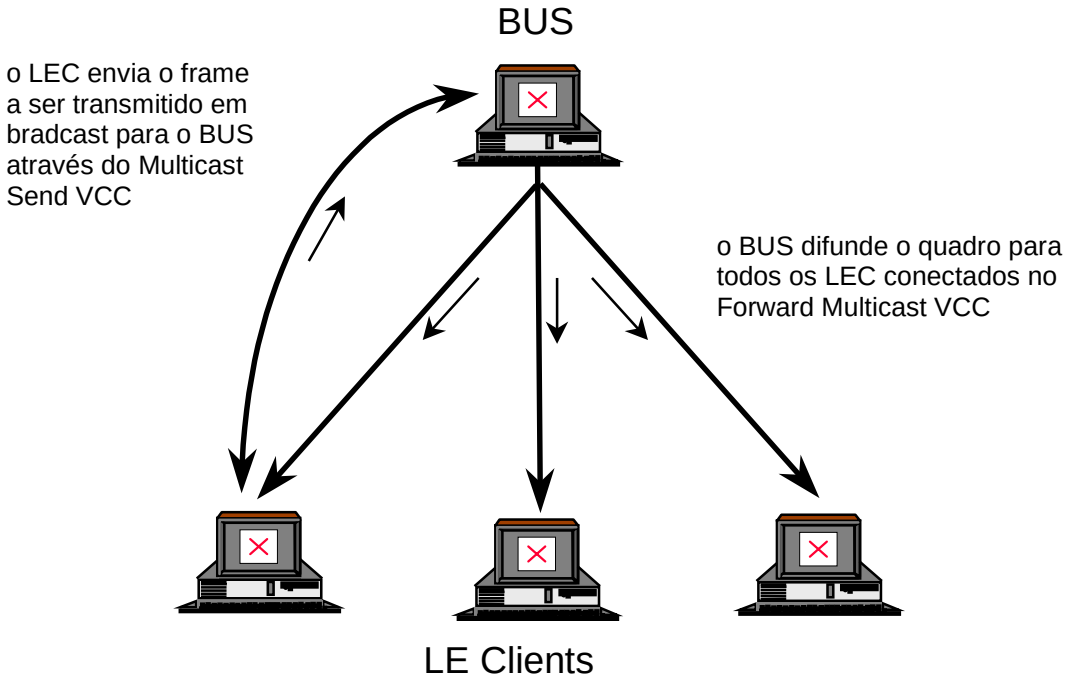


Figura 22: Mensagens broadcast e multicast gerenciadas pelo BUS.

Um LEC envia quadros de dados para o BUS o qual serializa os quadros e retransmite-os para um grupo de LECs participantes. A serialização é necessária para evitar que quadros AAL5 de fontes diferentes sejam intercalados.

- **LAN Emulation Configuration Server (LECS):**

  O LECS é responsável pela associação dinâmica de diferentes LECs a diferentes LANs emuladas e pela manutenção de uma base de dados contendo as associações resultantes. O LECS informa quais LECs fazem parte de quais LANs emuladas. Existe um LECS por domínio administrativo e este serve a todas as LANs emuladas que se encontram em seu domínio.

O protocolo LANE não especifica onde os componentes da LANE descritos acima devem ser localizados..

2.3.5 Tipos de Conexões

As conexões utilizadas pelo serviço LANE são de 2 tipos:

1. Conexões de controle: conectam o LEC ao LECS ou ao LES para troca de mensagens de controle para realização do serviço LANE:

- **VCC direto de configuração** (Figura 23)**:** é um canal virtual bidirecional ponto-a-ponto estabelecido entre o LEC e o LECS. Por este canal, o LECS envia

ao LEC a informação de configuração, incluindo o endereço do LES. A entidade pode manter ou não este VCC enquanto participa da LAN Emulada. É ativado pelo LEC como parte da fase de conexão ao LECS;

- **VCC de controle direto** (Figura 24)**:** é um canal virtual bidirecional ponto-a-ponto estabelecido entre o LEC e o LES para envio de tráfego de controle. Ele é ativado pelo LEC na fase de iniciação. Uma vez que o LES tem a opção de usar um caminho de retorno para enviar dados de controle para o LEC, isto requer que o LEC aceite o tráfego de controle deste VCC;

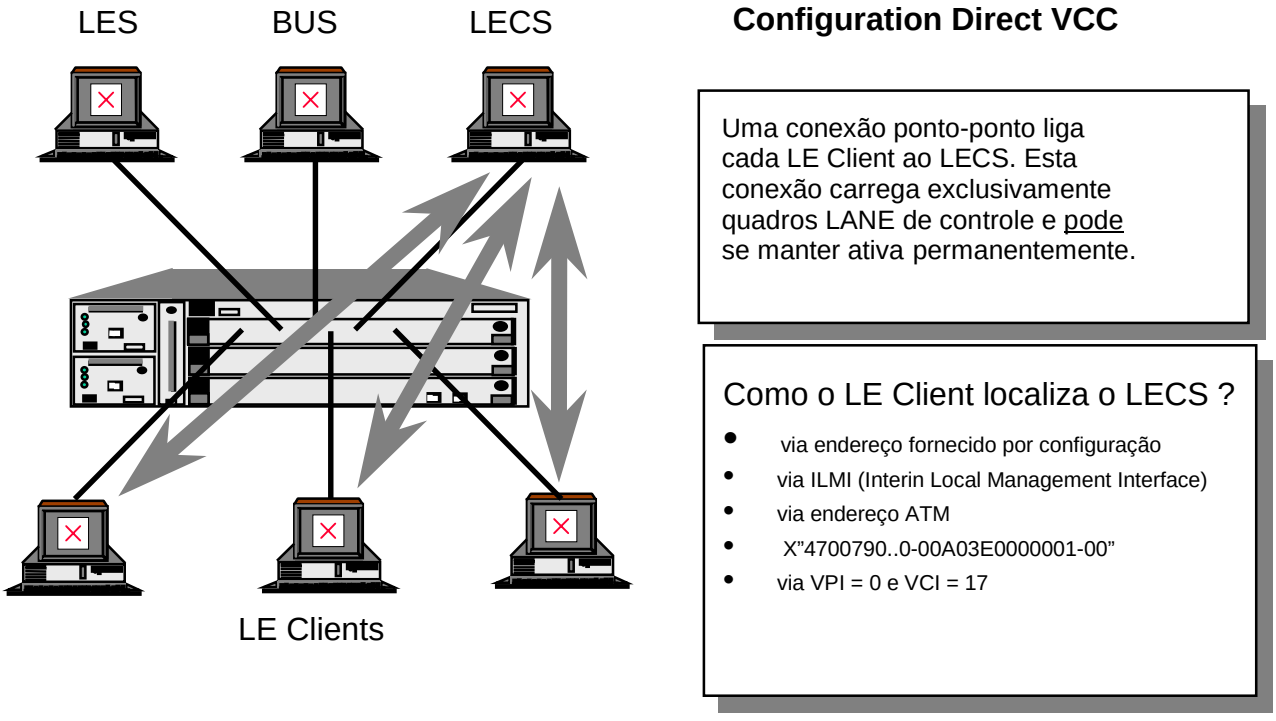
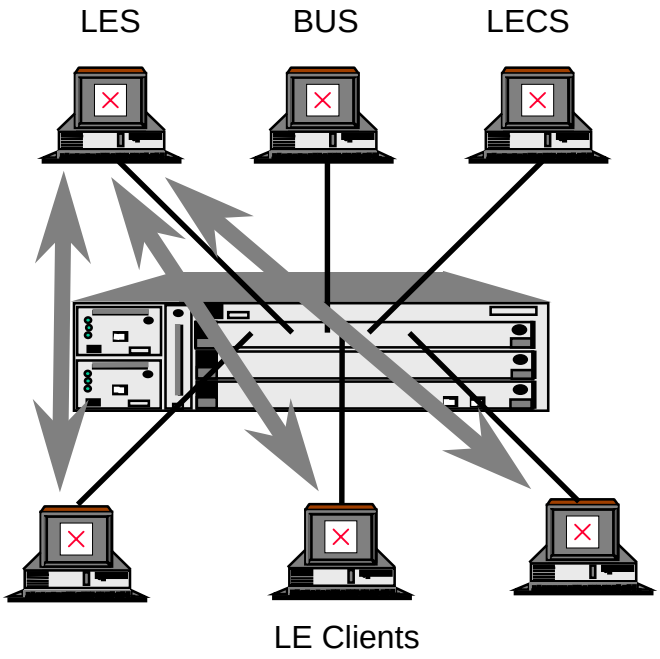


Figura 23: Conexão de controle entre o LEC e o LECS.

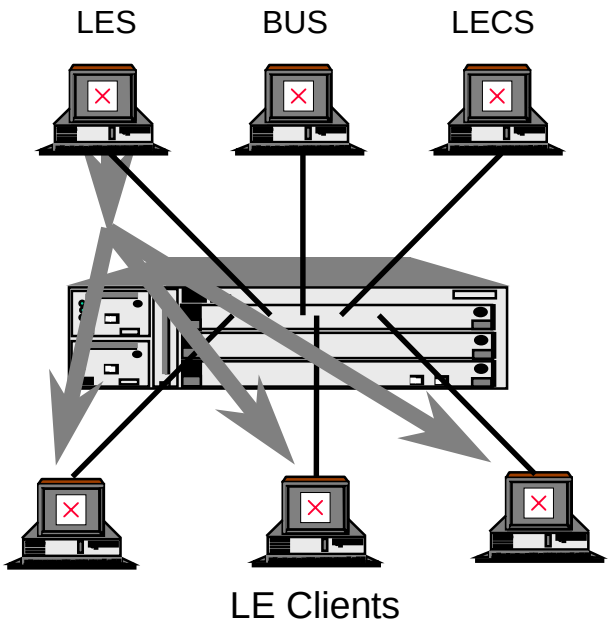


Control Direct VCC

Uma conexão ponto-ponto liga cada LE Client ao LES. Esta conexão carrega exclusivamente quadros LANE de controle e deve-se manter ativa permanentemente.

Figura 24: Conexão de controle entre o LEC e o LES.

- **VCC de controle distribuído** (Figura 25)**:** é um canal virtual unidirecional através do qual o LES retorna informações (tais como resoluções de endereço) ao LEC, via conexões ponto-a-ponto ou ponto-multiponto. Deve ser ativado pelo LES como parte da fase de iniciação.



Distributed Control VCC

Uma conexão ponto-multiponto opcional liga o LES a cada LE Client. Esta conexão carrega exclusivamente quadros LANE de controle.

Figura 25: VCC de Controle de Distribuído.

2. Conexões de dados: VCCs de conexões de dados conectam os LECs entre si e com o BUS. Suportam quadros Ethernet e Token Ring dependendo do tipo da LAN emulada.

- **VCC direto de dados** (Figura 26)**:** é um canal virtual bidirecional ponto-a-ponto que estabelece comunicação entre dois LECs que desejam efetuar troca de dados *unicast*. Dois LECs utilizarão o mesmo canal virtual direto de dados para transportar todos os pacotes transmitidos entre eles.

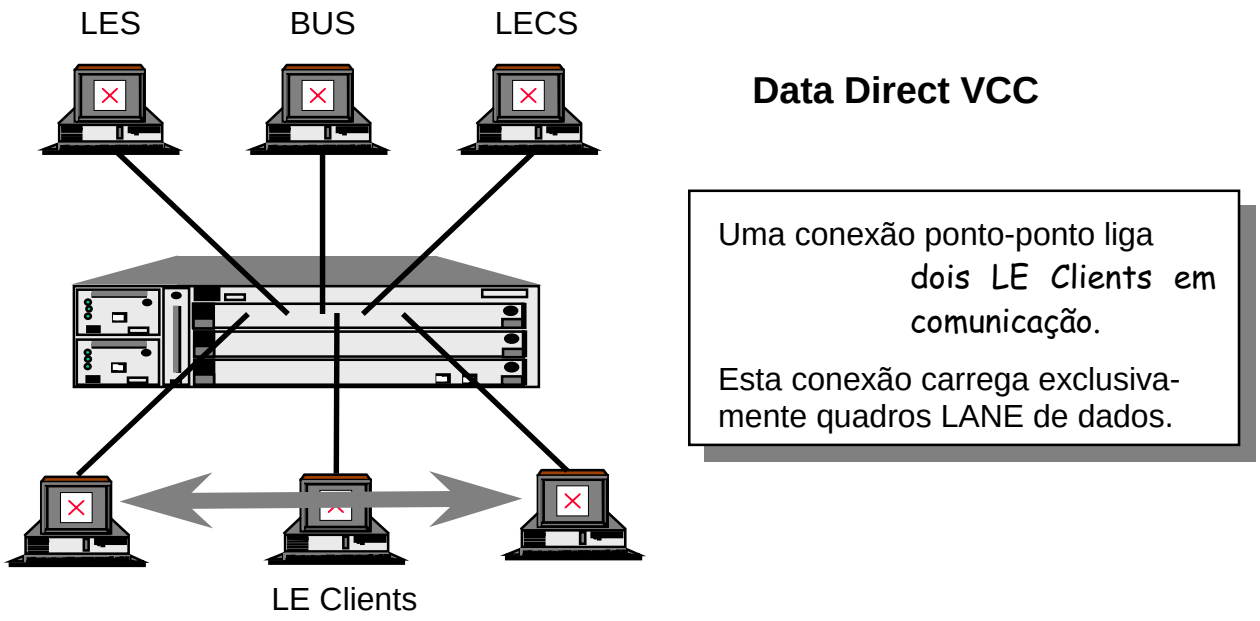


Figura 26: Conexões de dados para o LEC.

Quando um LEC tem um quadro para enviar para um endereço MAC onde o correspondente endereço ATM é desconhecido, o LEC deve gerar uma requisição LE_ARP para obter tal endereço. Uma vez que o LEC recebe uma resposta LE_ARP ele estabelece um VCC ponto-a-ponto (*Data Direct VCC*) se não estiver já estabelecido, sobre o qual mandará todos os dados subsequentes para aquele destino.

O LEC é responsável, após enviar uma requisição e receber uma resposta LE_ARP, por iniciar a sinalização para estabelecer este VCC direto de dados com o nome do cliente correspondeste ao LE_ARP.

Se o cliente não possui recursos suficientes para estabelecer um VCC direto de dados, ele deve enviar os quadros para o BUS até que o cliente consiga recursos para estabelecer um novo VCC direto de dados.

- **VCC de envio multicast** (Figura 27)**:** é um canal virtual bidirecional ponto-a-ponto estabelecido do LEC para o BUS. Este VCC é estabelecido usando o mesmo processo do VCC direto de dados. O LEC na fase de configuração envia um LE_ARP relativamente ao endereço de *broadcast* e, quando recebe uma resposta, inicia a sinalização para estabelecer este VCC bidirecional para o BUS.

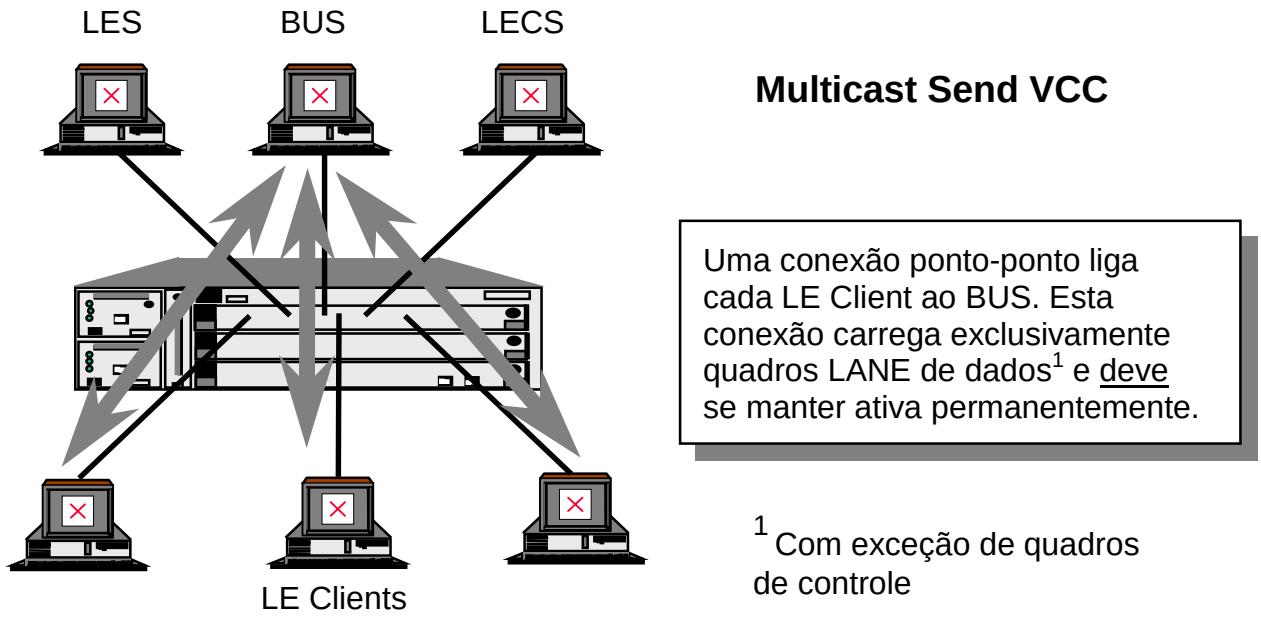


Figura 27: Conexões de dados para o BUS: Multicast Send.

- **VCC de Encaminhamento Multicast** (Figura 28)**:** é um canal virtual unidirecional do LEC para o BUS, tipicamente uma conexão ponto-multiponto, com cada LEC como uma folha. Depois do LEC estabelecer o VCC de envio multicast, o BUS inicia a sinalização para o VCC de Encaminhamento Multicast para o LEC. Este VCC é usado para distribuir dados do BUS. Ele pode ser uma conexão ponto-mutiponto, ou um VCC unidirecional ponto-a-ponto.

O VCC de Encaminhamento Multicast deve ser estabelecido antes do LEC participar da LAN emulada e deve ser mantido enquanto estiver participando desta. O BUS deve reenviar quadros para o LEC no VCC de envio multicast ou no VCC de Encaminhamento Multicast. Um LEC não receberá quadros duplicados enviados

por ambos os VCCs *multicas*t, mas deve estar apto a receber quadros de qualquer um dos dois.

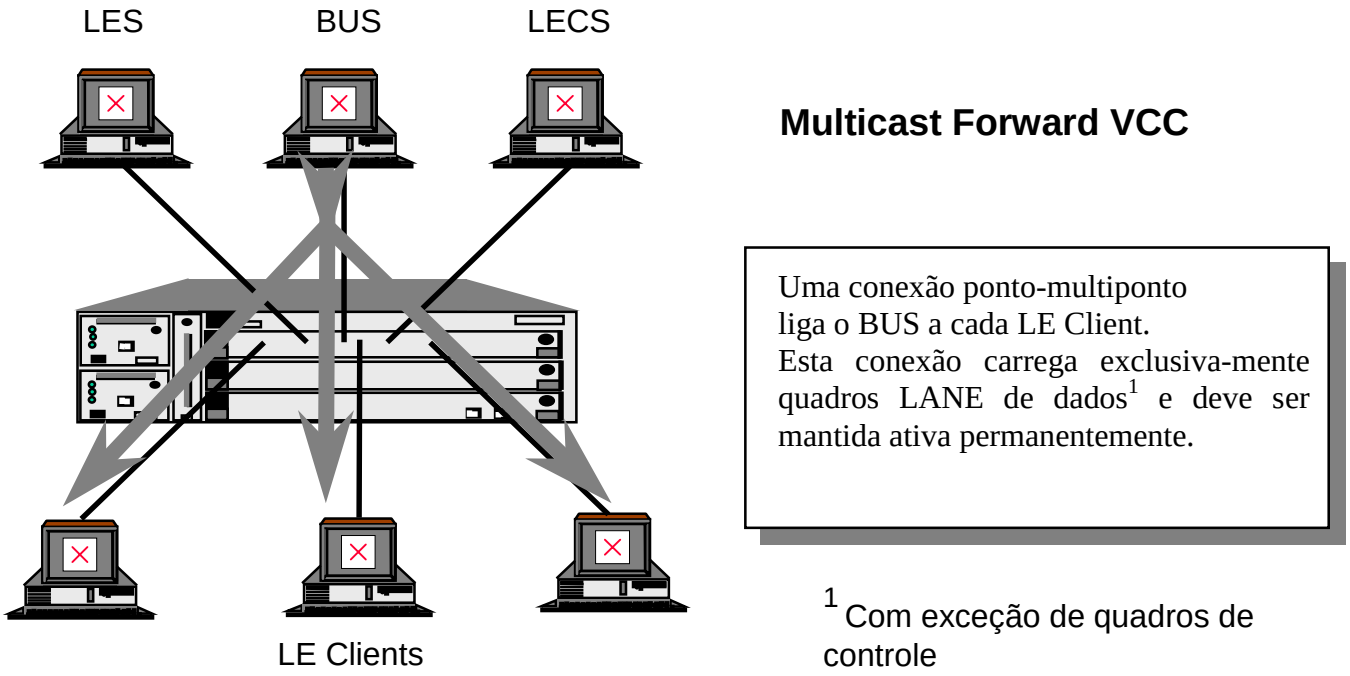


Figura 28: Conexões de dados para o BUS: Multicat Forward.

A Figura 29 ilustra esquematicamente as conexões do serviço LANE.

2.3.6 Funções do Serviço de Emulação de LAN

Os LECs passam por vários estados para participarem de uma LAN emulada. Esses estados (fases) indicam o progresso do LEC desde o momento em que o equipamento é ligado até o instante em que ele se encontra operacional.

2.3.6.1 Iniciação

Este item discute o estado Inicial, a função de configuração e a função de associação à LAN emulada. A iniciação é completada depois dos processos de

associação e registro inicial terem se completado e as conexões para o BUS sido estabelecidas. Neste ponto o LEC torna-se operacional (Figura 30).

- **Estado Inicial**

  No Estado Inicial existem parâmetros como o endereço, nome da LAN emulada, tamanho máximo de quadro etc., que são conhecidos pelo LES e LEC antes de participarem de uma função de configuração e uma função de fase de associação.

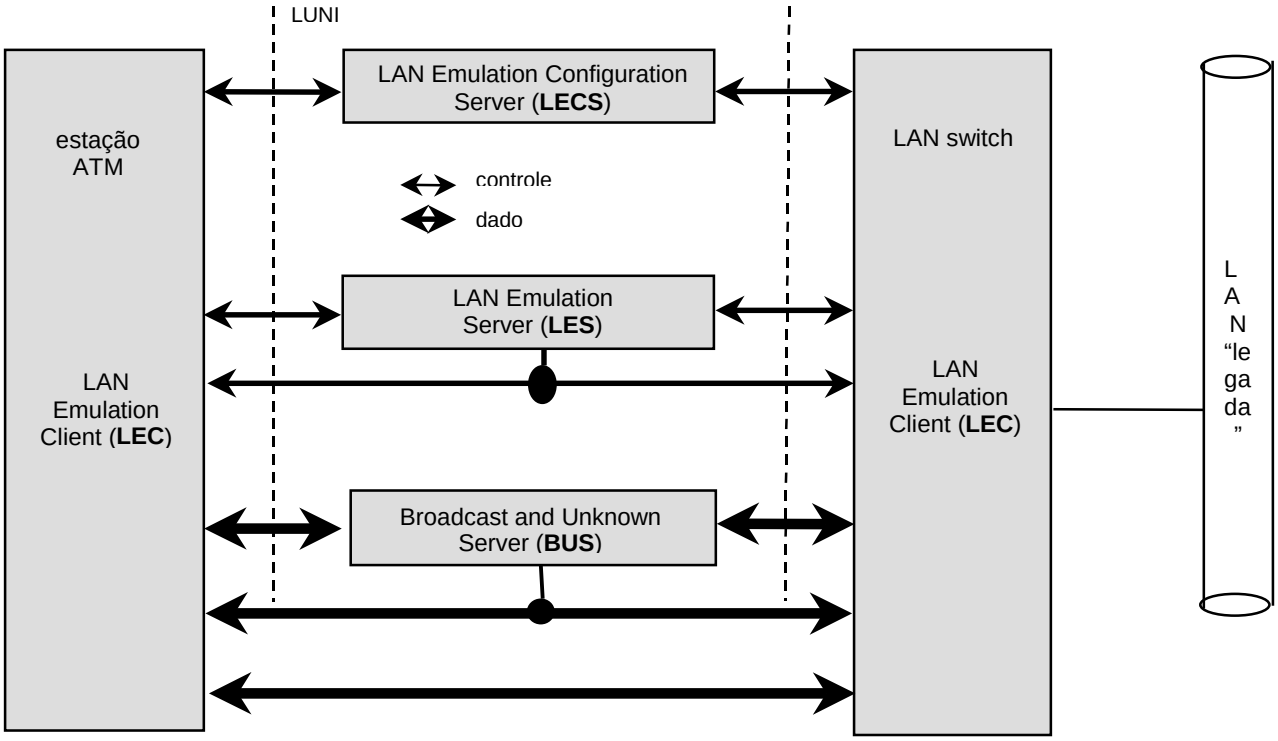


Figura 29: Conexões de dados e controle de uma LANE entre 2 LECs e o Serviço LANE.

- **Fases de Conexão ao LECS e Configuração**

  Para atingir operações de *plug-and-play* no caso dos LECs, o protocolo de LAN emulada requer um Servidor de Configuração (LECS). Um LEC descobre o endereço ATM do LECS e estabelece um VCC Direto de Configuração (*Configuration Direct VCC*) para o Servidor. Existem três maneiras para o LEC encontrar o endereço ATM do LECS:

- o endereço pode ser obtido do *switch* ATM ao qual o LEC está conectado via procedimento ILMI (*Interim Link Management Interface*);
- utilizando um endereço estabelecido pelo ATM Forum para o LECS;

- utilizando uma conexão permanente e conhecida (VPI=0 e VCI=17) em direção ao LECS.

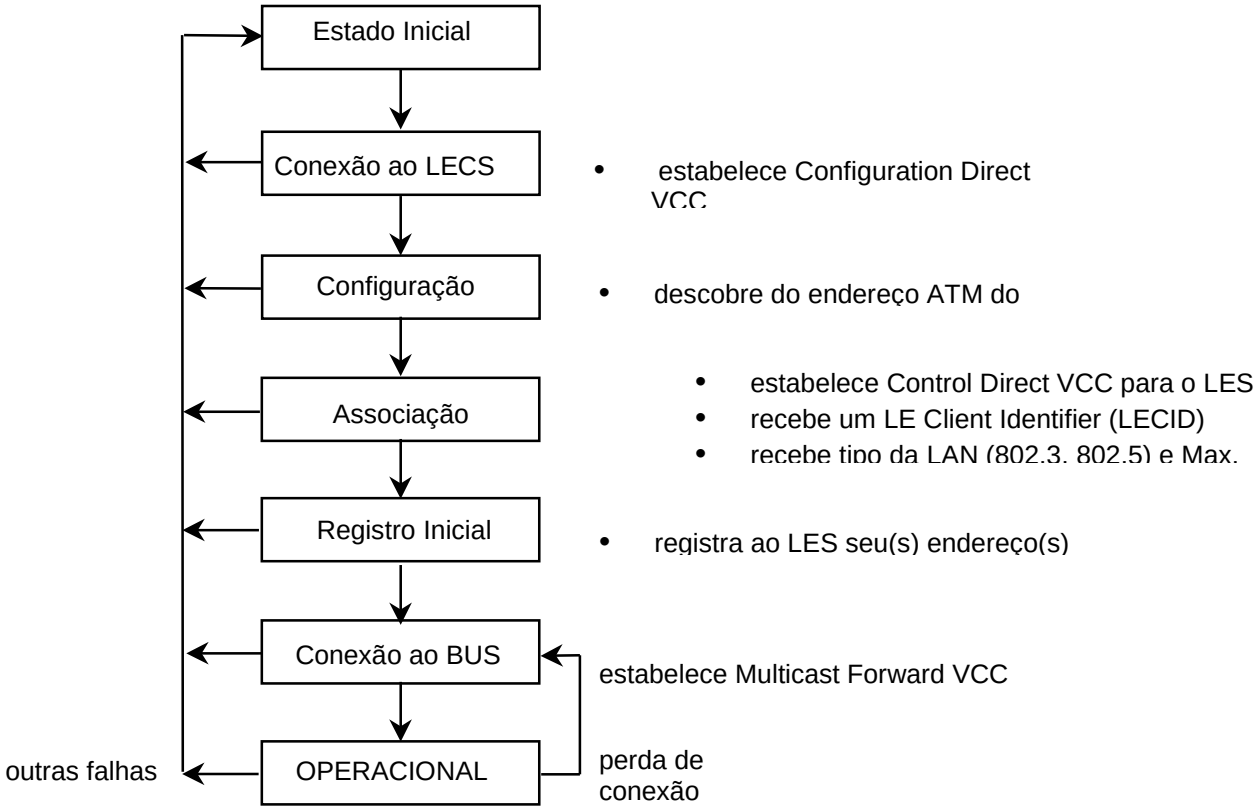


Figure 30: Inicialização, recuperação e terminação da ELAN.

Tendo estabelecido o VCC Direto de Configuração, o LEC envia um frame de controle de *Configure Request* para o LECS especificando pelo menos o endereço ATM do LEC. Opcionalmente, o LEC pode incluir um endereço MAC de 48 bits, um nome para a ELAN e/ou escolhas de tamanho máximo de quadro e tipo de LAN (IEEE 802.3/Ethernet ou IEEE 802.5/Token-Ring). O LECS usa parte ou todas estas informações para decidir a qual ELAN o LEC pode juntar-se e retorna o resultado em um *Configure Response.*

Um *Configure Response* com sucesso retorna ao LEC o endereço ATM do LES que atende aquela ELAN. A resposta pode também incluir informações adicionais

de configuração. Uma vez obtido o endereço do LES, o LEC pode opcionalmente liberar o VCC Direto de Configuração estabelecido com o LECS.

- **Fase de Associação**

O LEC usa o endereço ATM do seu LES para estabelecer um VCC Direto de Controle o Servidor e transmitir um quadro de *Join* Request contendo o seu endereço ATM, como também, uma indicação informando se o LEC irá agir como um *proxy* para endereços MAC não registrados (por exemplo, agir como uma ponte transparente IEEE 802.1D). O LEC pode também incluir um endereço MAC ou preferências de tipo de LAN e tamanho máximo de quadros.

O LES decide se o LEC pode juntar-se à ELAN. Se a requisição for concedida, o LES pode estabelecer um VCC Distribuído de Controle para o LEC. O LES então envia um Join Response para o LEC.

O LES apresenta considerável flexibilidade na construção do VCC Distribuído de Controle. As exigências são que cada LEC tenha exatamente um VCC Direto de Controle e não mais que um VCC Distribuído de Controle. Considere a **Error! Reference source not found.**O plano (a) pode ser implementado dentro de um *switch* ATM. Ele não tem nenhum VCC Distribuído de Controle; o LES deve copiar os quadros de controle que deseja enviar por *broadcast*. O plano (b) trabalha como o BUS, ou seja, todos os quadros de controle transmitidos pelo LES são enviados para todos os LECs. O plano (c) permite que o LES envie requisição de resolução de endereço MAC (LE_ARP - LAN *Emulation Address Resolution Protocol*) para endereços MAC não registrados somente para LECs proxy.

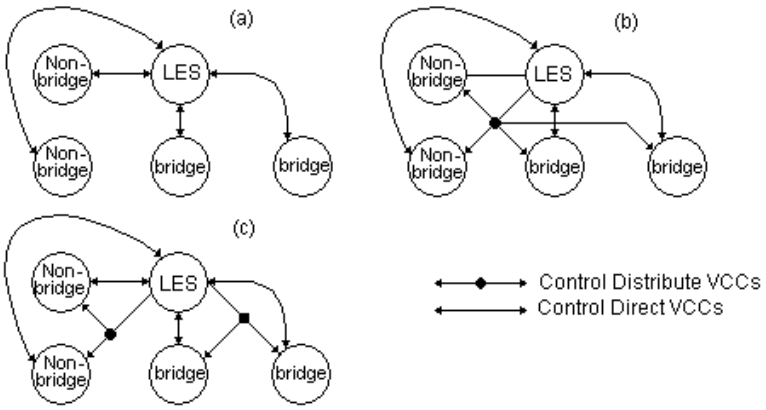


Figura 31: Possíveis arranjos de planos de controle.

Em qualquer destes planos, o LES pode otimizar o fluxo de requisição e respostas de LE_ARPs pelo envio destes para LECs específicos através do VCC Direto de Controle dos clientes.
Uma resposta de associação com sucesso diz ao LEC os valores corretos para o tipo da LAN, tamanho máximo do quadro e nome da ELAN. Também inclui um identificador de *LAN Emulation Client* de 16 bits (LECID). Este número é único entre todos os LECs pertencentes a uma dada ELAN. Existe um lugar para este identificador em todo quadro de dados e todo quadro de controle do tipo requisição feito pelo LEC depois de se juntar à ELAN.

- **Registro Inicial**

Depois da associação à LAN emulada, um LEC pode registrar qualquer número de endereço MAC (ou Descritores de Rotas). O endereço MAC pode também ser registrado na fase de Associação. O Registro Inicial permite a um LEC verificar a unicidade do seu endereço local antes de completar a iniciação e tornar-se operacional.

- **Conectando ao BUS**

Para obter o endereço ATM do BUS, o LEC envia uma requisição LE_ARP para o LES. A requisição LE_ARP contém o endereço MAC de *broadcast* (0xFFFFFFFFFFFF), o LECID e o endereço ATM do LEC. O LES deve responder a esta requisição retornando um *LE_ARP Response* para o LEC contendo o endereço ATM do BUS. De posse do endereço o LEC pode estabelecer o VCC de Envio Multicast (*Multicast Send VCC*) para o BUS. O BUS responde a esta criação de VCC estabelecendo o VCC de Encaminhamento Multicast (*Multicast Forward VCC*) de volta para o LEC. Os parâmetros passados no processo de estabelecimento destas conexões garantem que o LEC sabe qual VCC vem do BUS de forma que ele pode ter certeza de aceitar a chamada e só liberá-la quando desejar deixar a ELAN.

- **Estado Operacional**

Um LEC entra no estado operacional após completar os requisitos para ingressar numa LAN emulada.

2.3.6.2 Resolução de Endereço

Cada cliente da ELAN mantém uma tabela de mapeamento de endereços ATM, sendo cada entrada nesta tabela válida apenas por uma determinado período de tempo após o qual é descartada. Quando do envio de um pacote para uma dado endereço de LAN, o cliente consulta sua tabela afim de encontrar o endereço ATM

correspondente. Se o cliente não o possuir, contacta o LES enviando-o um pedido de resolução de endereço de LANE (LE_ARP), através da Conexão de Controle Direta. O LES tem então a tarefa de responder a este pedido de LE_ARP, havendo então duas possibilidades:

- o LES possui o endereço ATM correspondente ao endereço MAC e envia-o, ao cliente que o solicitou, através de uma mensagem resposta (LE_ARP_ RESPONSE);
- o LES não possui o endereço ATM, e envia a todos os clientes da árvore da ELAN que não possuem os endereços registrados junto ao LES, como é o caso das *bridges* do tipo proxy, um pedido de LE_ARP_REQUEST.

Um cliente qualquer de LANE, ao receber um pedido de LE_ARP, se souber (constar em sua tabela) o endereço ATM correspondente ao endereço MAC, deve enviá-lo ao LES através de uma mensagem de LE_ARP_RESPONSE. A *bridge*, se tiver acesso ao endereço destino, isto é, o mesmo constar em sua tabela, responde ao LES com o seu próprio endereço ATM. Esta resposta é enviada pelo ao LEC que enviou o pedido LE_ARP, ou ainda, o LES envia-a através da sua conexão VCC de Controle Distribuído a todos os LECs de modo que estes possam armazenar a informação caso necessitem resolver o endereço posteriormente.

2.3.7 Transferência de Dados

Em uma ELAN, os pacotes de dados (não de controle) são transmitidos através das conexões diretas entre clientes ou através das conexões que os clientes mantêm com o BUS. Quando um LEC estabelece, via mecanismo de resolução de endereço, que um determinado *host* destino corresponde a certo endereço ATM, e quando o cliente sabe que possui um VCC direto de dados para aquele endereço ATM, então um quadro endereçado para o *host* destino deve ser enviado via VCC direto de dados.

Em paralelo à resolução de endereços via LES, o cliente pode também encaminhar quadros de dados ao BUS, que envia-os para todos os clientes da ELAN. Isto tem por objetivo acelerar o processo de conexão com o destino, em virtude dos quadros poderem ter alguma exigência com relação a atrasos durante a transmissão. Para evitar-se o uso abusivo dos canais de *broadcast*, um cliente tem um número limite de quadros que pode enviar num determinado período de tempo.

Uma vez que o cliente de origem descobre o endereço ATM do cliente destino, estabelece uma Conexão Direta de Dados com o último, não mais enviando os dados pelo servidor BUS. Neste caso há dois caminhos para os dados unicast entre o cliente emissor e o destino, um através do BUS e outro através da Conexão Direta entre eles. Afim de garantir a correta sequência de chegada dos quadros no destino, o cliente de origem envia uma mensagem do protocolo de *flush* pelo caminho antigo (via BUS). Quando o cliente destino recebe a mensagem de *flush*, envia um sinal de que o caminho antigo está limpo e o novo está pronto para ser usado. Enquanto isso, o cliente de origem retém os quadros que quer enviar através da conexão virtual direta.

No caso de quadros *unicast* endereçados a um *host* cujo endereço não se encontra registrado no LES, os quadros transmitidos pelo BUS serão enviados para todos os clientes de sua árvore de *hosts* através de sua conexão ponto-multiponto de Distribuição Multicast. O endereço destino recebe então os quadros a ele destinados, ainda que esteja localizado após uma *bridge* do tipo *proxy* pois, neste caso, apesar da mesma não possuir em sua tabela de mapeamento o endereço requerido, transmite o quadro *unicast* para todos os integrantes do segmento de rede a que dá acesso. Sendo a *bridge* capaz de aprender os endereços através da interceptação dos quadros que por ela passam, uma resposta por parte do cliente destino faz com que a *bridge* adicione esta nova entrada à sua tabela, podendo responder a partir de então a um pedido de LE_ARP do LES acerca de tal endereço.

Os quadros *multicast* são enviados ao BUS para distribuição a todos os clientes conectados à malha da Conexão de Distribuição Multicast do BUS. Para um cliente receber quadros *multicast* deve estar conectado ao BUS através da Conexão de Envio Multicast, devendo receber os quadros transmitidos pelo BUS em qualquer destes dois canais virtuais. Esta decisão é deixada para o serviço de LAN Emulation, não para o cliente.

O cabeçalho de qualquer quadro de dados enviado de um cliente para o BUS deve conter o valor de LECID fixado para aquele cliente. É requerido ao BUS preservar o cabeçalho de um quadro retransmitido. Assim, o cliente pode identificar e filtrar quadros enviados comparando o campo LECID a seu próprio valor LECID

2.3.8 Formatos de Quadro

O serviço LANE utiliza quadros AAL5. Os quadros AAL5 ocupam um número inteiro de células ATM, 48 bytes de dados por célula, com a última célula contendo um *trailer* que inclui o número de octetos no quadro e um CRC para garantir a integridade dos dados.

Existem dois tipos de quadros utilizados no LAN Emulation: quadros de dados e quadros de controle.

Os quadros de dados contém ou o valor “0” ou o LECID do LEC nos dois primeiros octetos. Seguindo estes padrões estão os quadros simples IEEE 802.3/Ethernet ou IEEE 802.5/Token-Ring. Os quadros Token-Ring também incluem um *byte* de PAD e um *byte* de controle de quadro (FC). Nenhum teste de seqüência de quadro (FCS) é incluíd em quadros de dados. A LANE conta com o *checksum* da AAL5. Cada ELAN tem um tamanho máximo de quadro, e este valor é distribuído para cada LEC quando ele se junta à ELAN. Existem quatro valores de tamanho de quadro, 1516, 4544, 9234 e 18190 *bytes*. A escolha do tamanho máximo do quadro é independente do tipo de ELAN. É permitida uma Ethernet com tamanho máximo de quadro de 18190 octetos.

Os quadros de controle têm um formato comum. Os primeiros dois octetos contêm o valor X”FF00”, um valor ilegal para um LECID, para diferenciá-lo dos quadros de dados. Os campos de um quadro desse tipo podem ser observados na figura a seguir.

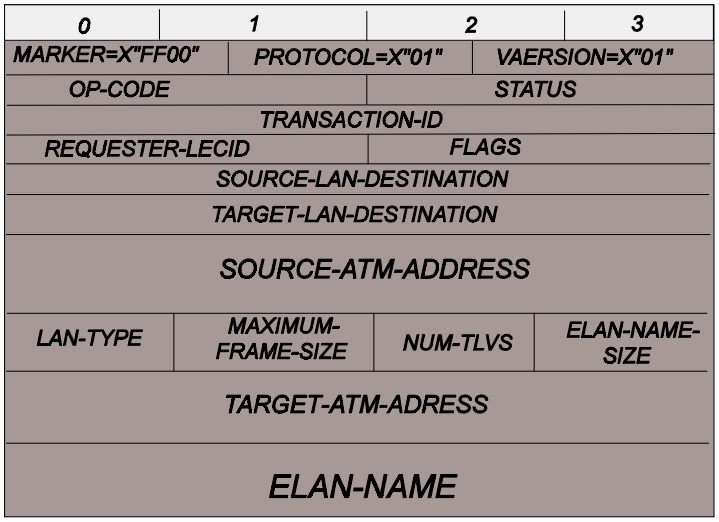


Figura 32: Frame de Controle.

2.3.9 Funções que não são providas pela LAN Emulation

A ELAN não resolve os problemas existentes para interconexão entre as tecnologias Ethernet e Token Ring. Para tanto, duas LANs emuladas distintas são interconectadas via roteadores ou pontes da mesma forma que duas LANs distintas são conectadas atualmente. LAN emulada também não permite que uma estação receba todos os quadros numa LAN lógica e não suporta protocolos da camada MAC existentes (gerenciamento SMT/Token), por exemplo, não suporta *Carrier Sense Multiple Access with Collision Detection (CSMA/CD)* para Ethernet ou passagem de token para Token Ring.

O protocolo LANE não provê emulação *Fiber Distributed Data Interface* (FDDI). Entretanto, o roteador ou um *switch* pode chavear tráfego FDDI sobre um serviço ATM LANE após a conversão dos pacotes Ethernet ou Token Ring.

2.3.10 Considerações Finais

Verificou-se que a proposta de LANE atende às necessidades dos usuários de redes locais conectados a uma rede ATM, fornecendo-lhes um serviço de *broadcast* e um ambiente semelhante ao de uma LAN tradicional, sem a necessidade de alteração

de qualquer dos protocolos de rede que estiverem em uso (IP, IPX, DECnet, Appletalk, etc.). A entidade de LANE encarrega-se de estabelecer as conexões de canais virtuais entre os clientes de origem e os de destino, ficando portanto a camada ATM transparente para a camada de rede.

Um aspecto deficiente da proposta da LAN emulada é a presença de roteadores entre as ELANs, criando uma estrutura rígida de transferência de dados entre *hosts* de ELANs distintas, e fazendo com que em uma rede de grande abrangência (WAN) haja um número desnecessário de VCCs para a comunicação entre clientes de ELANs distintas.

Outro aspecto a ser melhorado na proposta de LANE é o aproveitamento da característica da rede ATM de garantia de qualidade de serviço (QoS). O ATM possui várias classes de serviço (UBR, ABR, VBR, CBR) caracterizando o fluxo de dados conforme a sua prioridade entre os demais fluxos em trânsito pela rede. Já a Emulação de Rede Local utiliza apenas a categoria de *melhor esforço* (Unspecified Bit Rate - UBR), que foi idealizada para aplicações tolerantes a atrasos, não sendo própria, por exemplo, para aplicações de tempo real. Como a classe de serviço UBR não oferece garantias do serviço de tráfego, não são especificados parâmetros de QoS para a mesma. Há portanto uma exigência por parte de usuários e desenvolvedores de aplicações críticas como as de áudio e vídeo, de uma melhoria na proposta de LANE, resultando em uma pesquisa feita pelo próprio Fórum ATM, no sentido de definir um sistema que aproveite melhor tais recursos da tecnologia ATM. É nesta linha de pesquisa que surgiu o MPOA (*Multiprotocol Over ATM*), que endereça a questão do número excessivo de VCCs em uma rede composta por várias ELANs e utiliza parâmetros de qualidade de serviço da rede ATM em ELANs.

Outra proposta que está sendo decidida atualmente no âmbito do ATM Fórum diz respeito a LANE versão 2.0. Nesta proposta algumas das restrições comentadas anteriormente relativamente ao LAN Emulation deverão ser resolvidas como, por exemplo, a necessidade da utilização de roteadores para interação de *hosts* em redes emuladas diferentes, como também, a possibilidade da utilização da qualidade de serviço pelo ATM.

2.4 A proposta de Multi-protocolo sobre ATM - MPOA

A especificação MPOA consiste em uma estrutura de comunicação que permite a sobreposição de protocolos de rede (camada 3 do modelo OSI) sobre uma base ATM de transmissão de dados. Utiliza o protocolo de Emulação de Rede Local para o estabelecimento de VLANs do tipo IEEE 802.1 sobre a nuvem ATM e para comunicação com as redes locais tradicionais. Também utiliza o protocolo NHRP, com algumas modificações, para a resolução de endereços e conseqüente otimização da comunicação entre *hosts* ATM e demais clientes MPOA, através da criação de conexões virtuais (VCs) entre os mesmos - conexões estas chamadas de atalhos (*VC-shortcuts*). Tais conexões de atalho eliminam o gargalo causado pelo emprego de roteadores na interconexão entre VLANs, característica da emulação de rede local.

Como prevê o mapeamento da camada de rede diretamente sobre a camada ATM, o MPOA permite que protocolos como o RSVP (do IETF), que operam no nível da camada de rede, possam oferecer às aplicações de camadas superiores conexões com níveis diferentes de QoS, próprias do ATM.

MPOA implementa o conceito de "Virtual Router", ou técnica de roteamento virtual, que consiste na separação entre as tarefas de chaveamento/envio de pacotes e de roteamento, como ferramenta para a otimização do uso dos VCs e redução de tempos de atraso.

Diferentemente das propostas do IETF que procuram resolver o problema do IP tendo redes NBMA como suporte, o MPOA procura resolver qualquer protocolo de nível 3 especificamente sobre o ATM.

2.4.1 Descrição do MPOA

O Serviço MPOA tem por meta oferecer à camada de rede representada por diversos protocolos, uma base de comunicação sobre uma nuvem ATM. Tais protocolos de camada 3 podem estar rodando em *hosts* conectados diretamente à nuvem ATM, *hosts* em redes locais tradicionais ou operando com Emulação de Rede Local (LANE).

O MPOA consiste de servidores, clientes MPOA e fluxos de informação entre estes componentes.

2.4.1.1 Componentes MPOA

O serviço MPOA é constituído de dois componentes básicos: cliente MPOA (**MPC**) e Servidor MPOA (**MPS**). A estrutura dos elementos é mostrada na Figura 33.

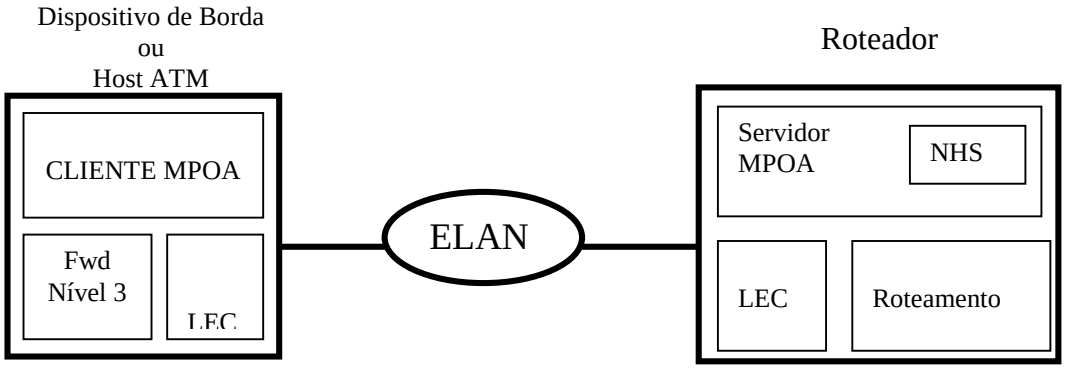


Figura 33: Componentes do Sistema MPOA

Podemos observar a partir da figura 33 que o cliente MPOA é parte de dois tipos de dispositivos: dispositivo de borda utilizado para interconectar redes locais herdadas (*legacy LANs*) e *hosts* ATM. O Servidor MPOA é incluido no roteador tendo como co-residente o servidor NHS (*Next Hop Server*).

A função principal do MPC é enviar e receber informações nos atalhos estabelecidos entre dois clientes MPOA (*VC-shortcuts*). Na realização desta função o MPC encaminha pacotes do protocolo de nível 3 sem, no entanto, ser responsável pela execução de protocolos de roteamento. O MPC ao detectar na entrada para a nuvem ATM um fluxo de pacotes encaminhado através do serviço LANE, ao roteador onde encontra-se o MPS, ele utiliza o protocolo NHRP (*Next Hop Resolution Protocol*) para tentar determinar um atalho para o destino daquele fluxo. Estabelecido o atalho, o cliente passa a encaminhar diretamente os pacotes.

Do ponto de vista do fluxo de pacotes na saída do sistema MPOA (nuvem ATM) para aplicações executando em um *host* ATM ou em redes convencionais, o MPC recebe pacotes de outro MPC para serem encaminhados aos seus usuários/ interfaces locais. Os pacotes recebidos pelo MPC em seu atalho são acrescidos do DLL (*Data Link Layer*) adequado e enviados para a interface LAN, por exemplo a porta de uma *bridge*. A informação a ser utilizada, relativamente ao DLL adequado, é fornecida pelo MPS ao cliente MPC e armazenada na *cache* de saída deste último.

Um cliente MPOA pode atender um ou mais LECs e comunicar-se com um ou mais MPSs. A Figura 34 ilustra um MPC em um dispositivo de borda.

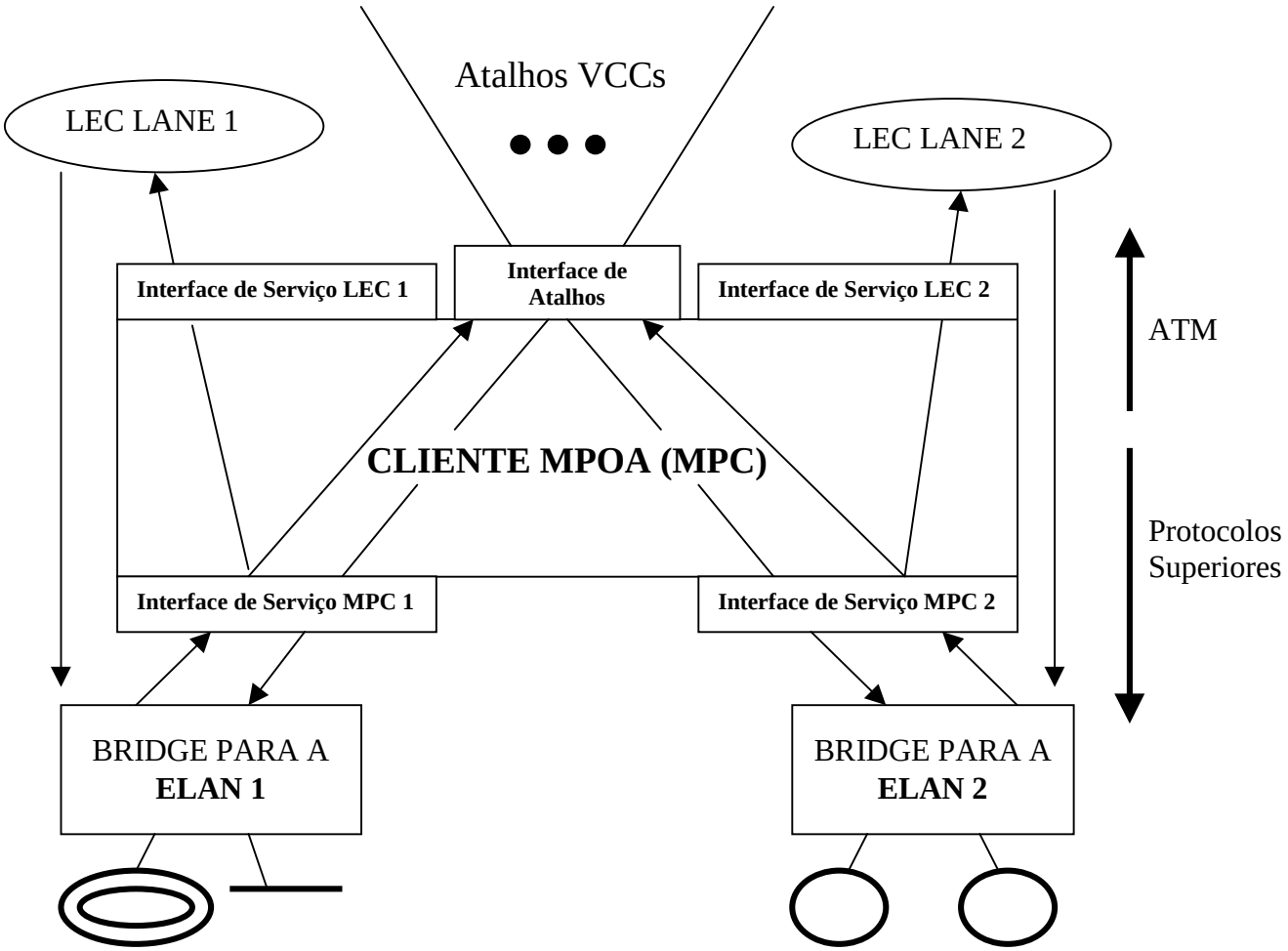


Figura 34: Exemplo de um MPC em um dispositivo de borda.

O Servidor MPOA (MPS) é um componente lógico presente no roteador que tem como objetivo fornecer informações de encaminhamento no nível 3 para os clientes MPOA. O MPS inclui o NHS (*Next Hop Server*) acrescido de algumas extensões à proposta NHRP. O MPS interage com o NHS local e o roteador quando da consulta realizada pelo MPC no caso de um fluxo de entrada, e fornece informações sobre o encapsulamento DLL ao MPC relativamente ao fluxo de saída.

Um roteador pode possuir um MPC co-residente ao MPS o que possibilitará o encaminhamento/roteamento de pacotes e o estacbelecimento de atalhos na nuvem ATM para encaminhamento dos fluxos detectados.

2.4.1.2 Fluxos de Informação

Os fluxos de informação no MPOA são classificados como fluxos de controle e fluxos de dados. Todos os fluxos de informação, com exceção dos fluxos de

configuração, utilizam VCs ATM e o encapsulamento LLC/SNAP [RFC 1483], assim como, a estrutura estabelecida no LAN Emulation (LANE).
A Figura 35 ilustra os vários fluxos de informação no MPOA.

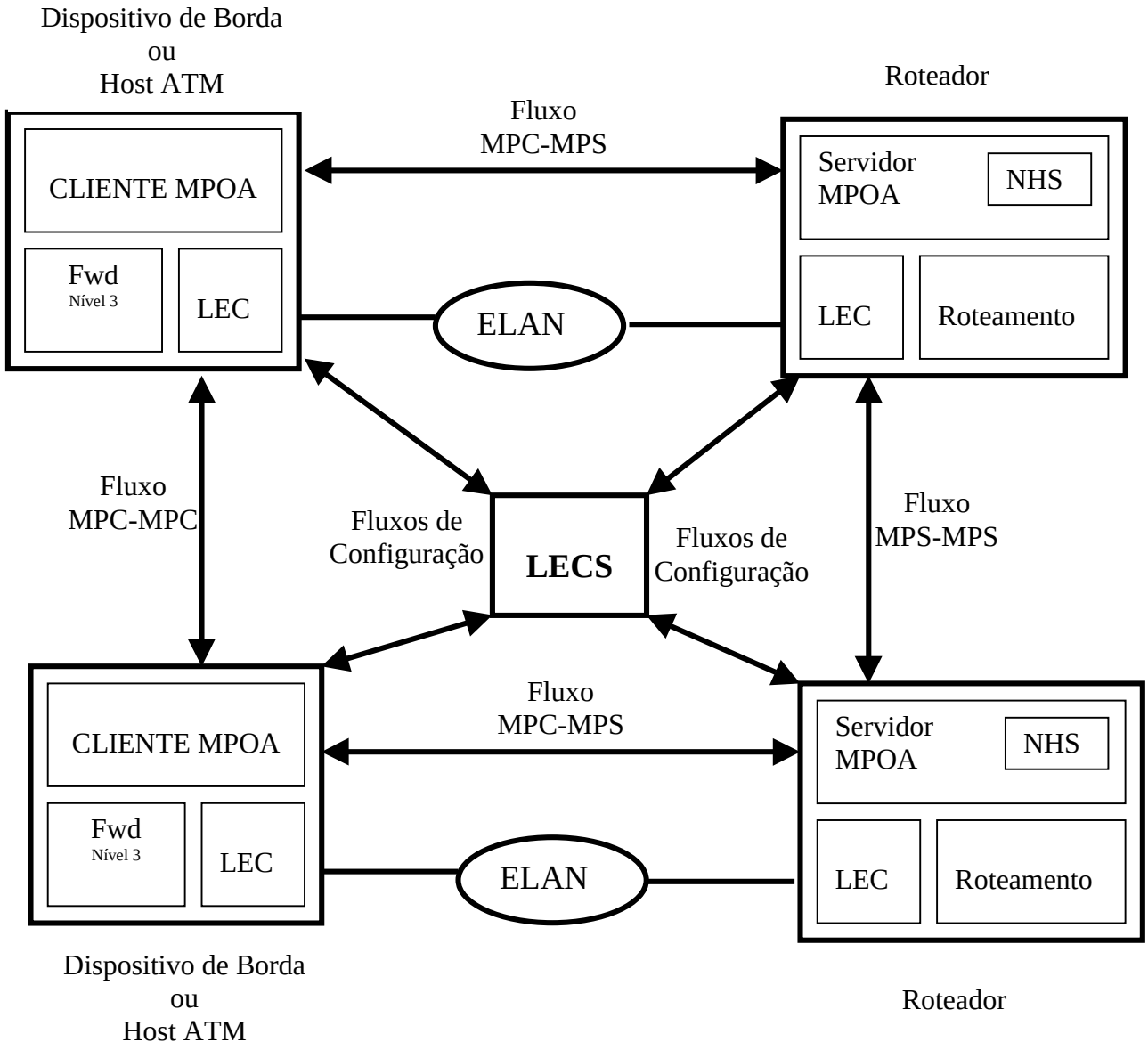


Figura 35: Fluxos de Informação no Sistema MPOA

Fluxos de Controle

- Fluxos de Configuração: informações trocadas pelos MPCs e MPSs com o Servidor de Configuração de LAN Emulada (LECS);

- Fluxos de Controle MPC-MPS: utilizado pelo MPC ou MPS para requisição e resposta de informações de encaminhamento que farão parte da *cache* de entrada do MPC. O MPS pode estimular (*trigger*) o MPC a realizar uma requisição através do envio, pelo MPS ao MPC, da mensagem *Trigger Message*. O MPC e o MPS podem enviar mensagem do tipo *Purge* quando detectarem que existe uma informação inválida na *cache* de entrada do cliente;

- Fluxos de Controle MPS-MPS: utilizado pelos protocolos de roteamento tradicionais e pelo NHS. O MPOA não define protocolo MPS-MPS;

- Fluxos de Controle MPC-MPC: um cliente pode enviar a outro cliente uma mensagem do tipo *Purge* quando ele detectar que recebeu um pacote inadequado (ex., direção errada) do enviado pelo cliente na outra extremidade do atalho. Isto permite que este último invalide a mensagem na sua *cache* de entrada.

Fluxos de Dados

- Fluxo MPC-MPC: utilizado basicamente para transferência de dados entre MPCs através de VCCs que servem de atalho no MPOA;

- Fluxos MPC-NHC: um cliente MPC pode enviar dados *unicast* a um NHC (*Next Hop Client*) e um NHC pode enviar dados *unicast* a um MPC;

- Outros fluxos de dado: outros fluxos podem ser encaminhados via roteadores através de interconexões envolvendo LANE, IP-Clássico e NHRP (Figura 36).

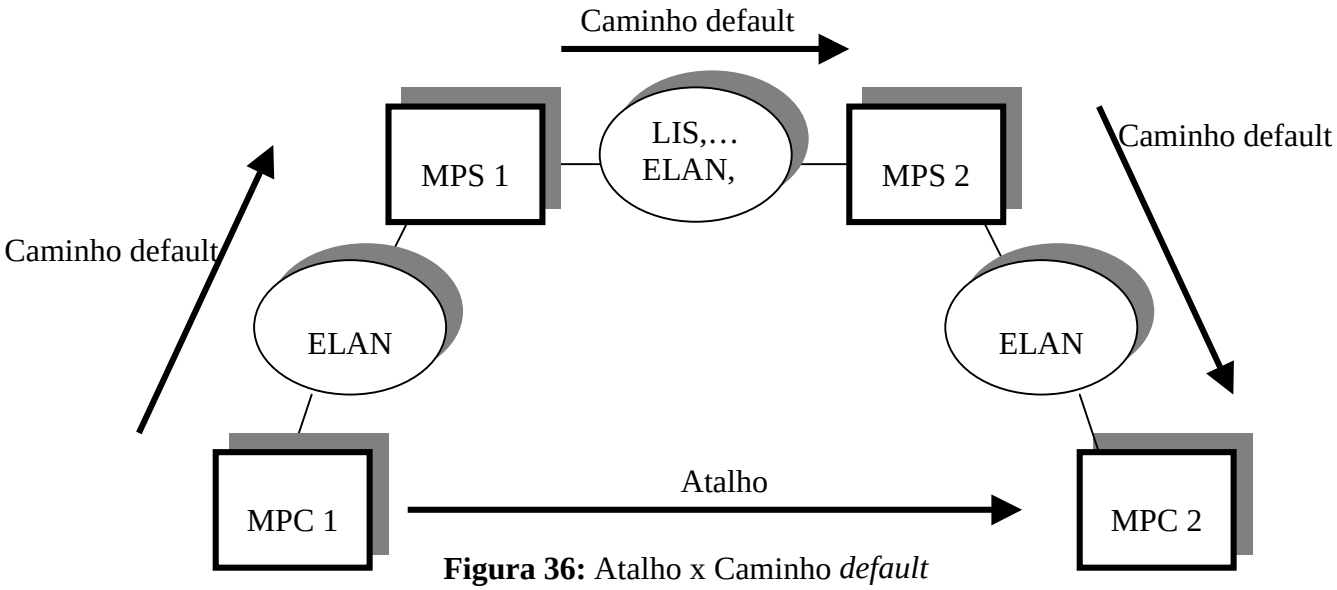


Figura 36: Atalho x Caminho *default*

2.4.1.3 Operação do MPOA

O MPOA realiza as seguintes operações:

- Configuração: possibilita a obtenção das informações de configuração;
- Descoberta: os MPCs e os MPSs aprendem sobre a existência de cada um deles;
- Resolução: determinação das informações de encaminhamento relativamente a um destino específico;
- Gerenciamento de conexões: corresponde à criação, manutenção e terminação de VCs para transferência de informações de controle e dados;
- Transferência de dados: encaminhamento das informações de nível 3 através de um VC-atalho.

Na fase de configuração os componentes do MPOA obtêm parâmetros de configuração através do LECS. A conexão com o LECS é feita na forma como definido pelo protocolo LANE. O componente MPOA deve enviar uma requisição de configuração para cada um dos LECs associados. A requisição de configuração utilizada pode ser idêntica à enviada pelo LEC (cliente LANE) com a adição do tipo (TLV)[6] do dispositivo, isto é, indicando se o dispositivo é um cliente ou servidor MPOA. Através desta informação o LECS retorna somente as informações relevantes para aquele tipo de componente.

Na fase de descoberta, os componentes MPOA utilizam uma extensão do protocolo LE_ARP do LANE indicando o tipo do dispositivo MPOA, ou seja, se é um MPC ou MPS e o endereço ATM.

Para a resolução do endereço, o MPOA emprega uma extensão do NHRP que permite a um cliente determinar o endereço ATM do ponto final para a criação de um atalho. O MPC, ao receber um quadro destinado a um MPS de uma das suas LANs emuladas, inicia um procedimento de deteção de fluxo. Durante este procedimento, os pacotes que compõem o fluxo são encaminhados ao roteador *default* via LANE, e a partir do roteador segue um caminho *default* na forma *hop-by-hop*. Quando o MPC fica convencido de que o tráfego deve ser beneficiado com um atalho, ele inicia o procedimento de resolução de endereço através do protocolo NHRP. A resposta pode conter, além do endereço ATM do MPC destino, informações adicionais como, por exemplo, o encapsulamento/tag a ser utilizado no VC do atalho.

O NHRP permite que a resposta seja enviada diretamente ao NHC que originou a consulta. Entretanto, o MPOA requer que a resposta seja retornada ao MPS de entrada, ou seja, ao primeiro MPS que recebeu a requisição de resolução de endereço enviada pelo MPC origem.

[6] TLV - *Type. Length, Value*.

Quando o MPS que serve o MPC destino da requisição NHRP recebe a requisição NHRP, o MPS envia uma mensagem MPOA *Cache Imposition Request* ao MPC destino. Esta mensagem fornece informações de estado e encapsulamento necessárias ao MPC destino. Este último responde ao MPS que o serve com a mensagem MPOA *Cache Imposition Reply* indicando se o MPC está apto a aceitar uma nova conexão. Deve ser observado que este MPS corresponde ao NHRP oficial no caso do protocolo NHRP.

Na fase de transferência de dados o MPOA tem como objetivo último uma transferência eficiente de dados *unicast*. A transferência dos dados possui dois modos de operação: fluxo *default* e fluxo via atalho. O fluxo *default* segue o caminho roteado (*hop-by-hop*) através da nuvem ATM. Neste caso o dispositivo ATM age como uma *bridge* no nível 2. O atalho, como já vimos, é estabelecido usando a resolução do MPOA e o mecanismo de gerenciamento da *cache*.

Quando o MPC deve enviar um pacote para o qual ele possui um atalho na direção do destino, ele age como um dispositivo de nível 3 e encaminha o pacote no atalho.

- Exemplo de uma rede empregando MPOA

Pode-se explicar os fluxos de dados entre componentes de um sistema tomando-se um exemplo de configuração de rede, contendo duas ELANs: ELAN1 e ELAN2. Cada ELAN contém um ou mais dispositivos de borda e *Hosts* ATM, e o dispositivo de borda suporta um ou mais *hosts*. A Figura 37 ilustra a arquitetura da rede a ser referenciada nos exemplos.

Os fluxos internamente a uma ELAN originam-se de um *host* MPOA ou de um *host* em uma LAN e fluem para um *host* ATM ou um *host* LAN na mesma ELAN. Estes fluxos utilizam o protocolo LANE para resolução de endereço e transferência de dado.

A Figura 38 mostra os fluxos intra-ELAN envolvendo as seguintes interações:

- Host 1MPOA para Host 2 MPOA – Figura 39;
- Host 1 MPOA para Host em LAN H 20 – Figura 40;
- Host em LAN H 10 para Host 2 MPOA – Figura 41;
- Host em LAN H 10 para Host em LAN H 30 – Figura 42.

A Figura 43 mostra os fluxos inter-ELAN envolvendo as seguintes interações:

- Host 1 MPOA para Host 5 MPOA – Figura 44;
- Host 1 MPOA para Host em LAN H 50 – Figura 45;
- Host em LAN H 10 para Host 5 MPOA – Figura 46;
- Host em LAN H 10 para Host em LAN H50 – Figura 47.

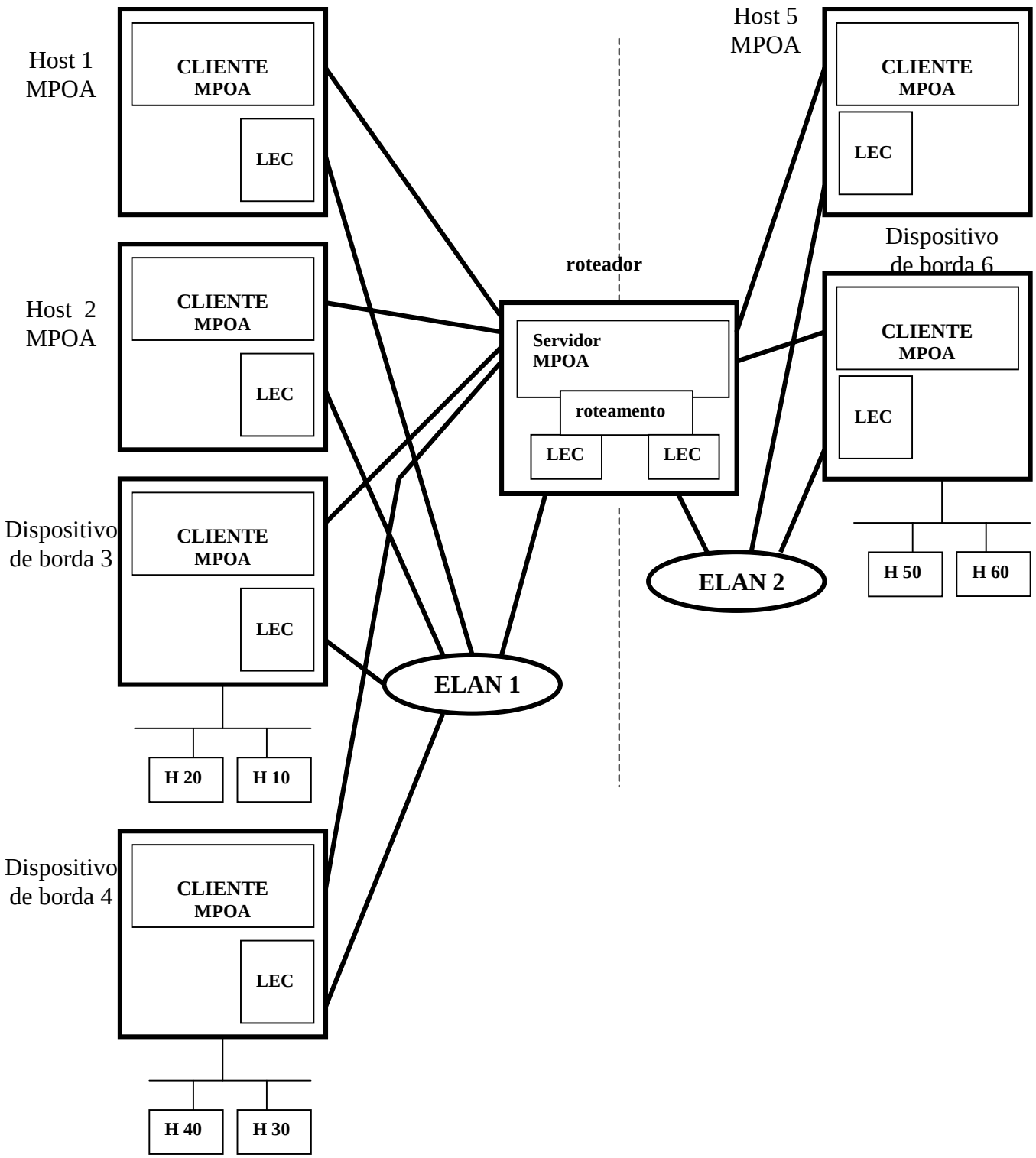


Figura 37: Exemplo de Configuração de Rede contendo duas ELANs

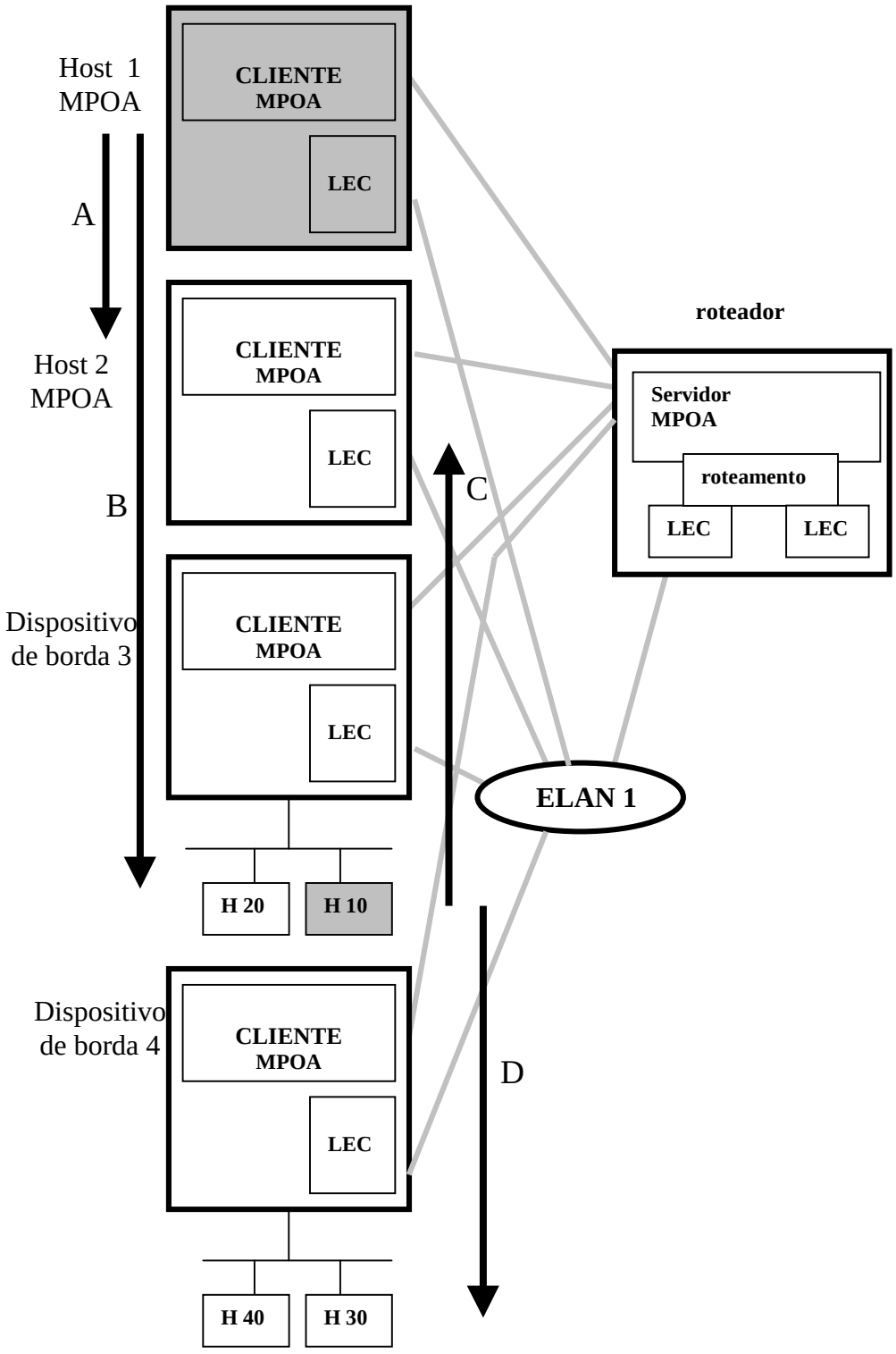


Figura 38: Fluxo Intra-ELAN**.**

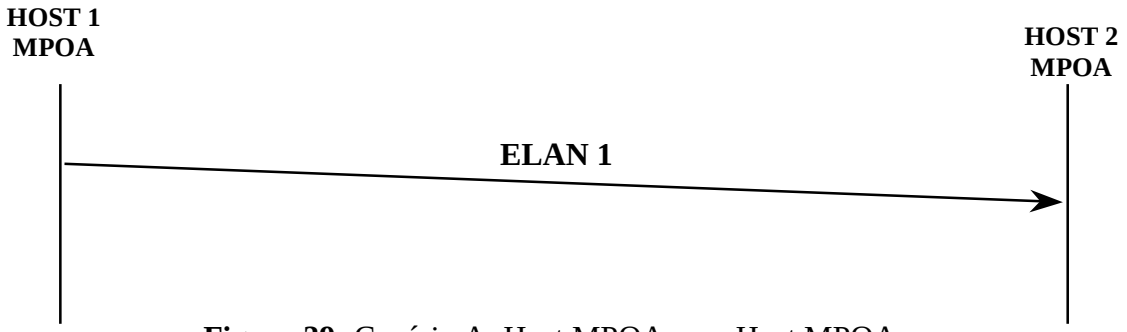


Figura 39: Cenário A: Host MPOA para Host MPOA.

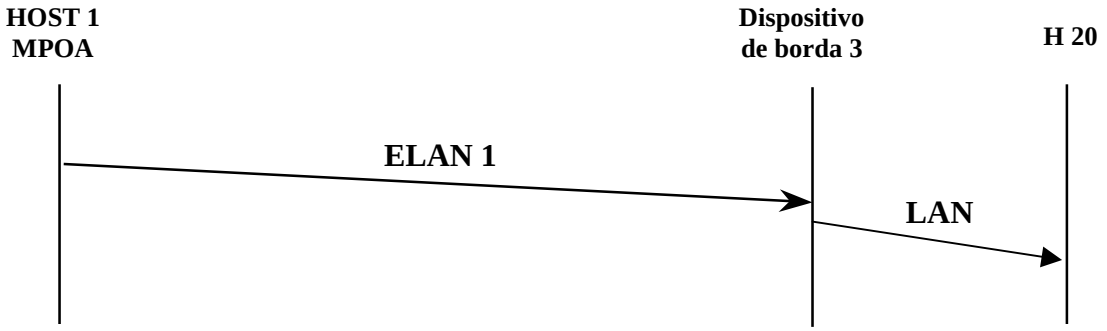


Figura 40: Cenário B – Host MPOA para Host em LAN

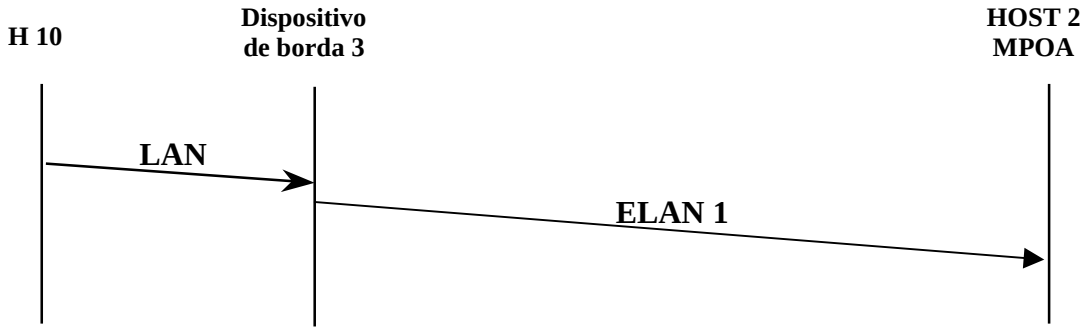


Figura 41: Host em LAN para Host MPOA.

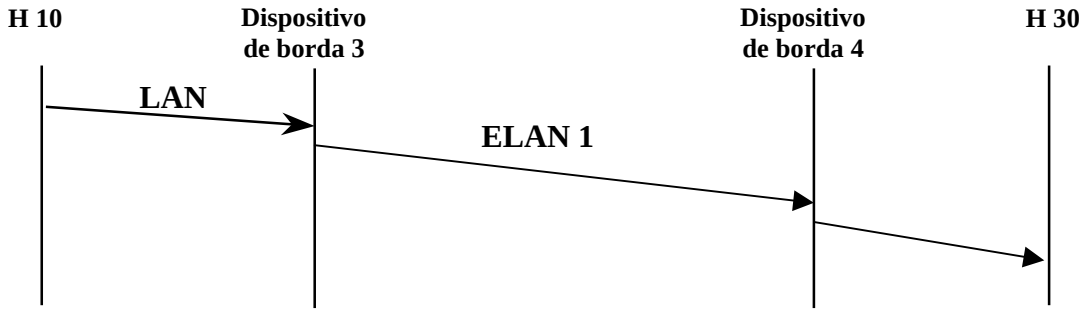


Figura 42: Host em LAN para Host em LAN.

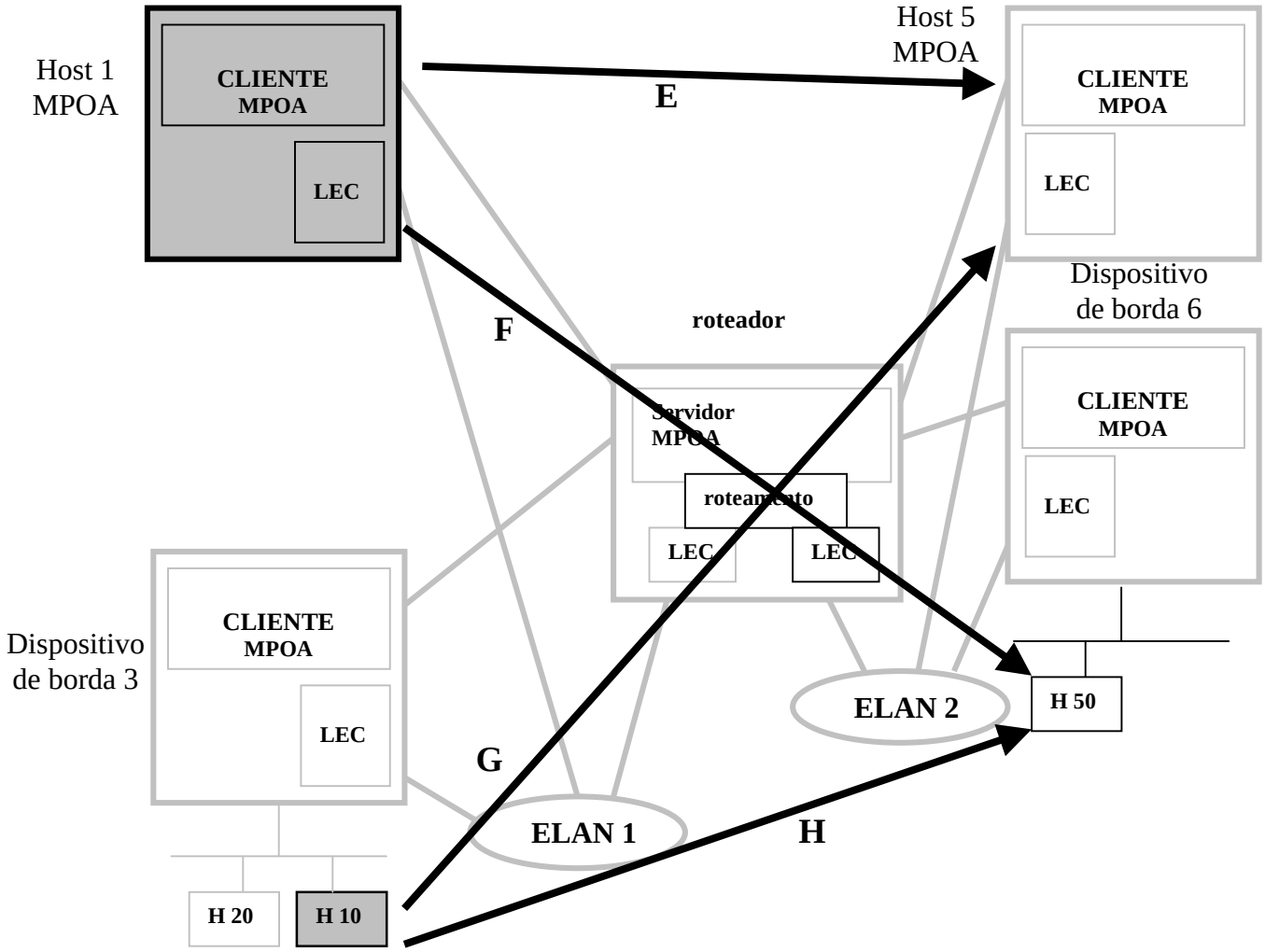


Figura 43: Fluxos entre LANs emuladas.

- ***Cenário E***: inicialmente o Host 1 envia pacotes através dos quadros da ELAN 1 para o roteador através de um *Data Direct VCC*. O roteador encaminha o pacote através da ELAN 2 para o Host 5 via *Data Direct VCC* (Figura 43) .Quando o Host 1 detecta um fluxo para o Host 5 ele envia uma requisição NHRP para o MPS de modo a obter o endereço ATM do Host 5. O roteador envia a mensagem *Cache Imposition Request* para o Host 5 o qual responde ao MPS indicando que ele aceita a conexão. O roteador retorna a resposta NHRP ao Host 1 contendo o endereço ATM do Host 5. O Host 1 atualiza a sua *cache* de entrada e estabelece uma conexão com o Host 5.

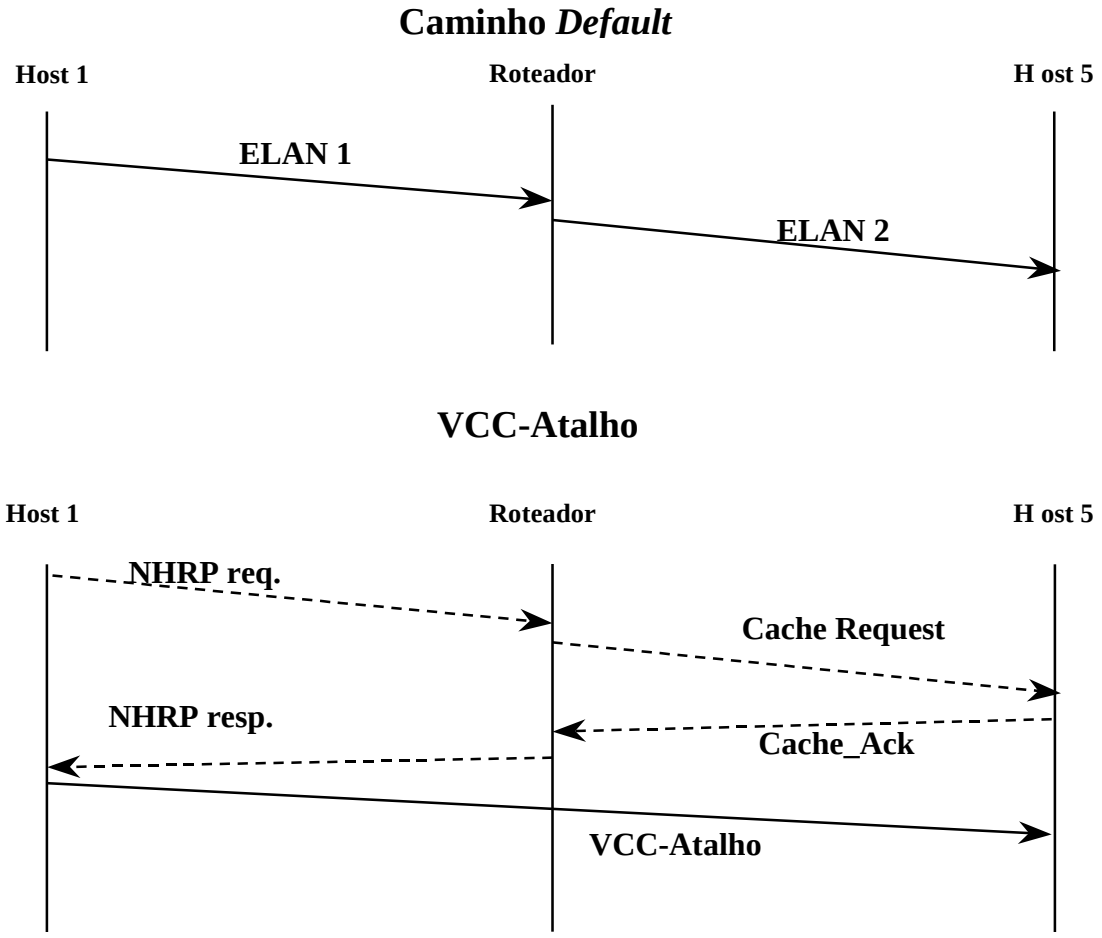


Figura 44: Cenário E - Host MPOA para Host MPOA.

- ***Cenário F***: O host 1 envia pacotes através dos quadros da ELAN 1 para o roteador via Data Direct VCC. O roteador encaminha o pacote em quadros da ELAN 2 através de um Data Direct VCC para o dispositivo de borda 6 o qual encaminha o pacote dentro de um quadro da LAN o qual é recebido pelo host H 50. Quando o Host 1 detecta um fluxo na direção do H 50 ele envia uma requisição NHRP ao MPS com o objetivo de obter o endereço ATM correspondente, no caso o endereço ATM do dispositivo de borda 6. O roteador consulta a *cache* do dispositivo de borda 6 o qual responde aceitando a conexão. A resposta NHRP retorna ao Host 1 com o endereço ATM do dispositivo de borda 6. A partir deste momento o Host 1 pode atualizar a sua *cache* de entrada e estabelecer uma conexão do dispositivo de borda 6 (Figura 45).

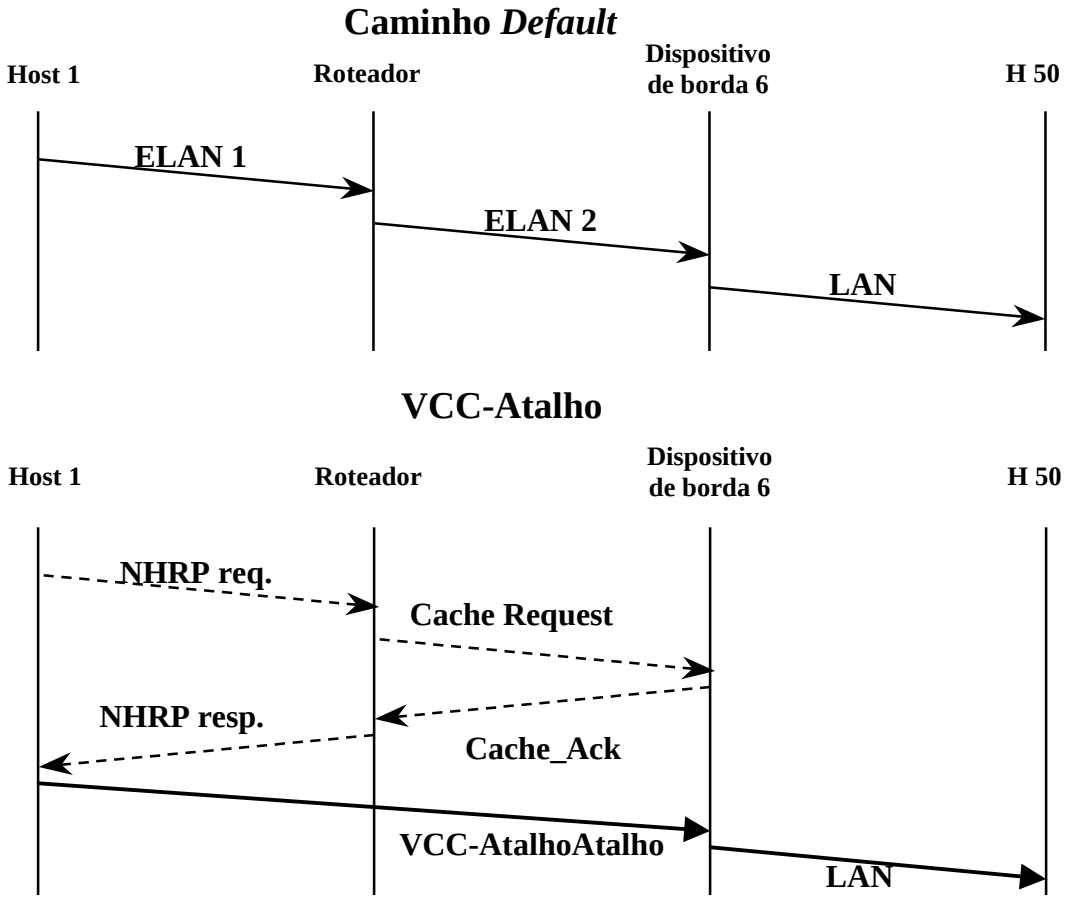


Figura 45: Cenário F - Host MPOA para Host em LAN.

Os cenários G e H encontram-se nas figuras 14 e 15, respectivamente, e a interpretação das figuras segue o mesmo esquema das anteriores.

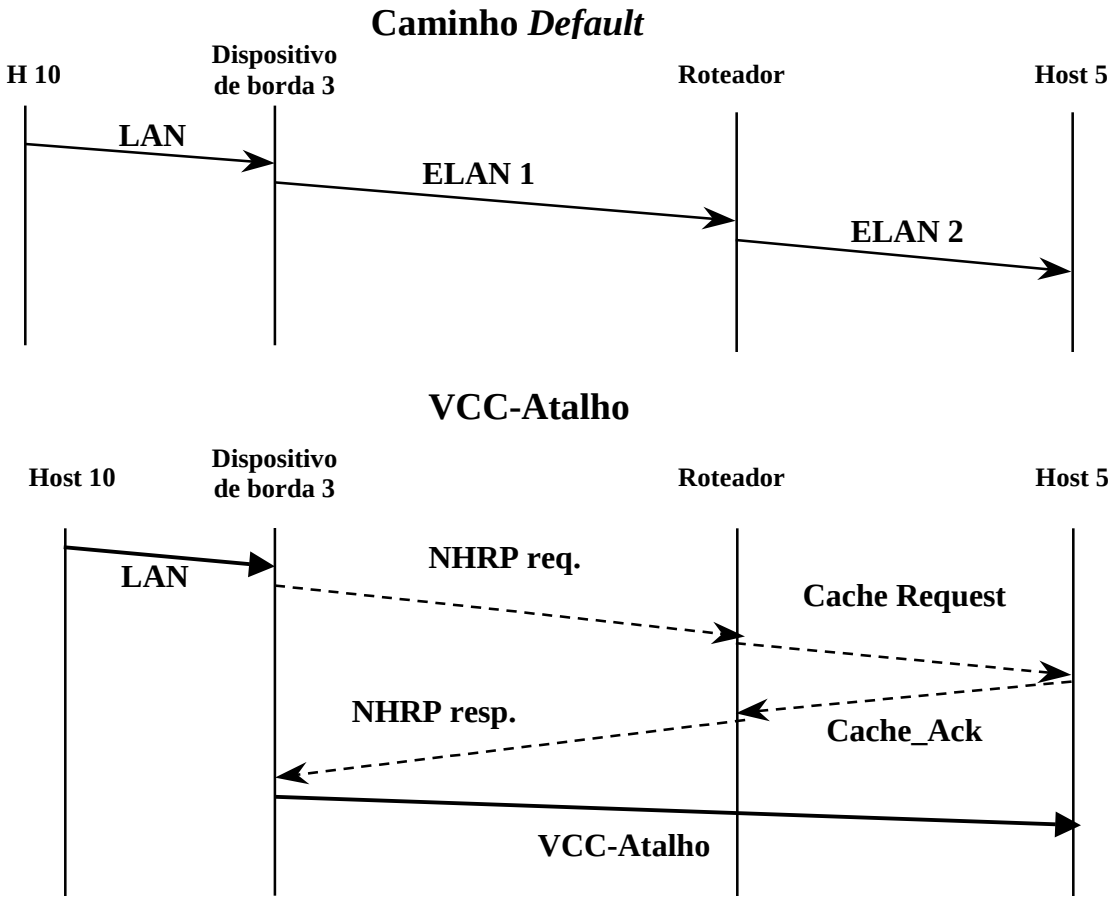
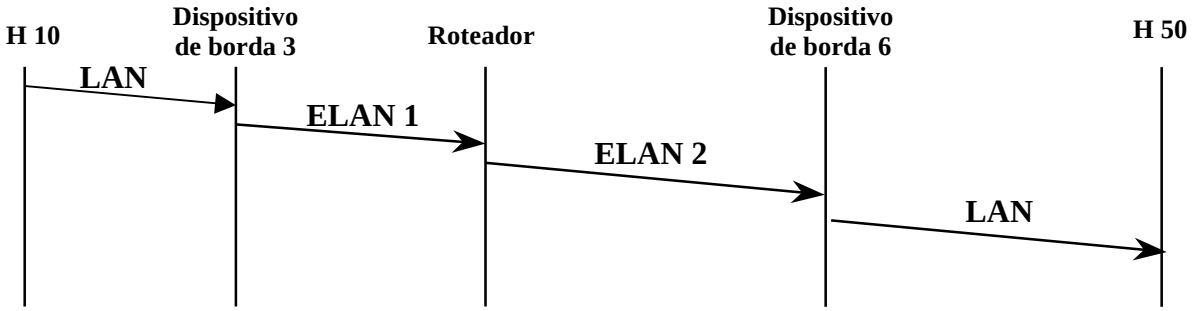


Figura 46: Host em LAN para Host MPOA.

Caminho *Default*



VCC-Atalho

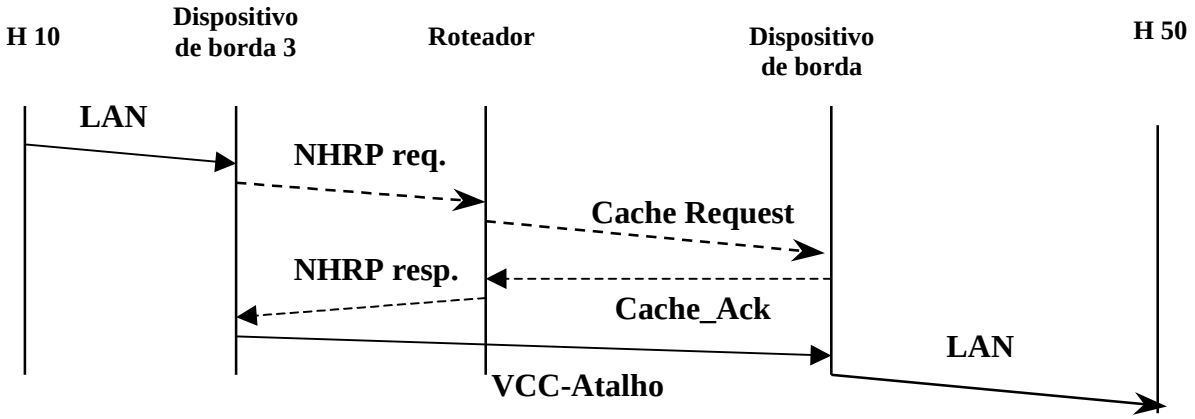


Figura 47: Host em LAN para Host em LAN

3 SOLUÇÕES SEGUNDO 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 as chaves ATM como elementos de rede (*peers*) dotando-as de certas funcionalidades da camada 3, principalmente roteamento no nível de rede. O modelo *peer* tem como motivação principal a elimininação da duplicidade de funções, notadamente as funções de roteamento. Existem algumas estratégicas para promover essa integração:

1. substituir o endereçamento, roteamento e sinalização do ATM Forum por outros associados a determinado protocolo de rede (IP, por exemplo). Esta é a solução adotada no IP Switching da Ipsilon Networks e TAG Switching da Cisco Systems;

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).

A primeira opção será explorada nesta seção. 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 às dos 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 dos protocolos atuais de roteamento por outro bastante recente. A aceitação desta solução também é incerta.

3.1 Considerações Sobre Roteamento

O protocolo IP, por operar sem conexão, exige dos roteadores uma 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[7] 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).

[7] Roteador conectado ao *backbone* central da rede.

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 necessidade de diminuir o atraso na comutação de pacotes motivou a introdução da tecnologia ATM com sua elevada capacidade de comutação e baixo atraso. Um primeiro passo é a substituição da malha de roteadores no *backbone* central por uma malha de chaves ATM, formando uma "nuvem ATM". Seja uma configuração tradicional onde uma nuvem ATM conecta roteadores (denominados roteadores de borda, Figura 48. Nesta topologia, supondo uma solução segundo o modelo *overlay*:

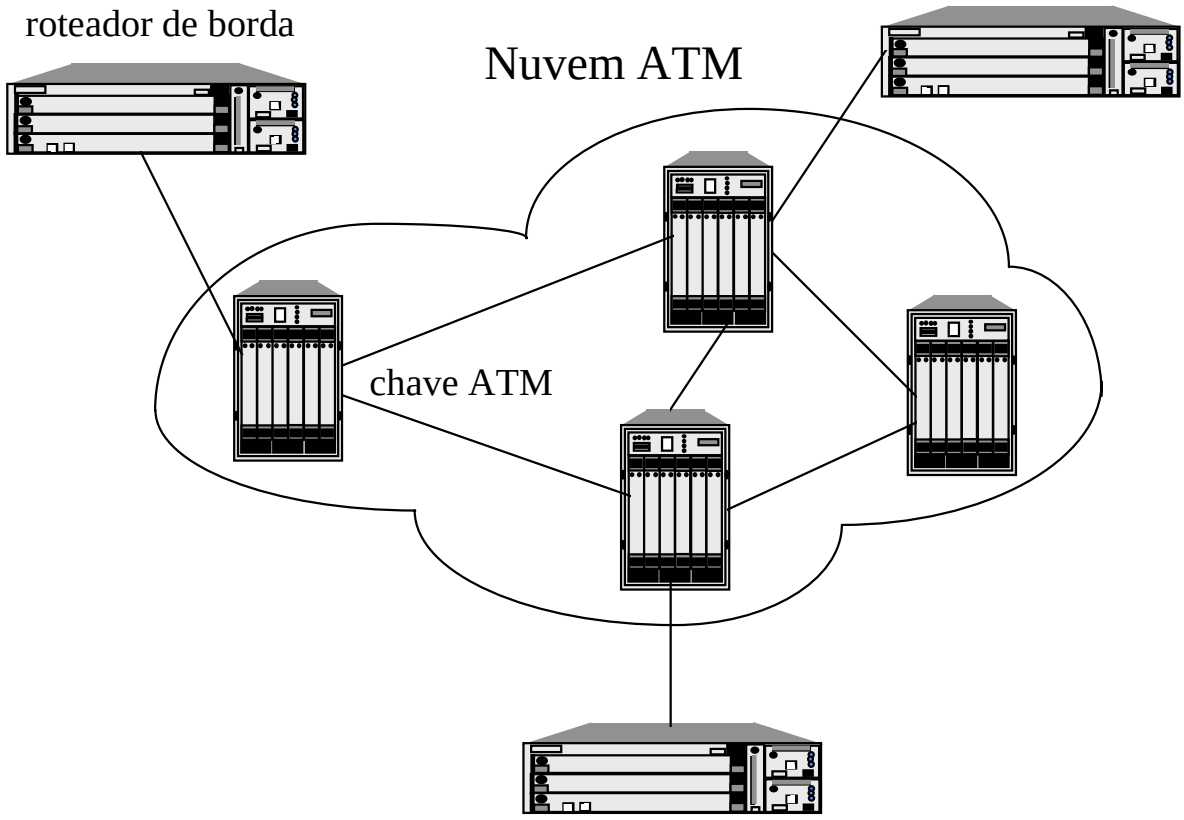


Figura 48: Roteadores de borda conectados por nuvem ATM.

- 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;
- a nuvem ATM se apresenta como uma infra-estrutura de camada 2 para os protocolos de rede.

Para redes *routed* 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 com o 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).

As soluções no âmbito do modelo *peer* procuram contornar os problemas acima.

3.2 O Conceito de Fluxo

Uma estratégia para o modelo *peer* se baseia no conceito de 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 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

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:
  - ◊ enderecos IP de origem e de destino;
  - ◊ protocolo de transporte;
  - ◊ ports de origem e destino (definidos na PDU de transporte).

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, 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.

A importância do conceito de fluxo advém do fato que, uma vez estabelecido um fluxo, os pacotes subsequentes podem utilizar o mesmo roteamento dos antecessores. Em outras palavras, fluxos eliminam a necessidade de roteamento "por pacote", diminuindo o atraso na propagação de pacotes.

No caso da utilização de tecnologia ATM, fluxos podem ser associados a canais virtuais (pares VPI/VCI). Após estabelecido um fluxo, o dispositivo ATM pode simplesmente chavear as células que compõem[8] o fluxo sem necessidade de roteamento no nível de rede. Entretanto, soluções nesta linha devem contemplar a possibilidade de alteração de rota para fins de controle de congestionamento ou falha de *links*. Em geral, soluções que exploram o conceito de fluxo estabelecem um fluxo por um período curto (1 segundo), restabelecendo-o (eventualmente por outra rota) no caso de sua continuidade.

A Figura 49 ilustra uma chave ATM adaptada para operar como *peer* de rede e explorando o conceito de fluxo.

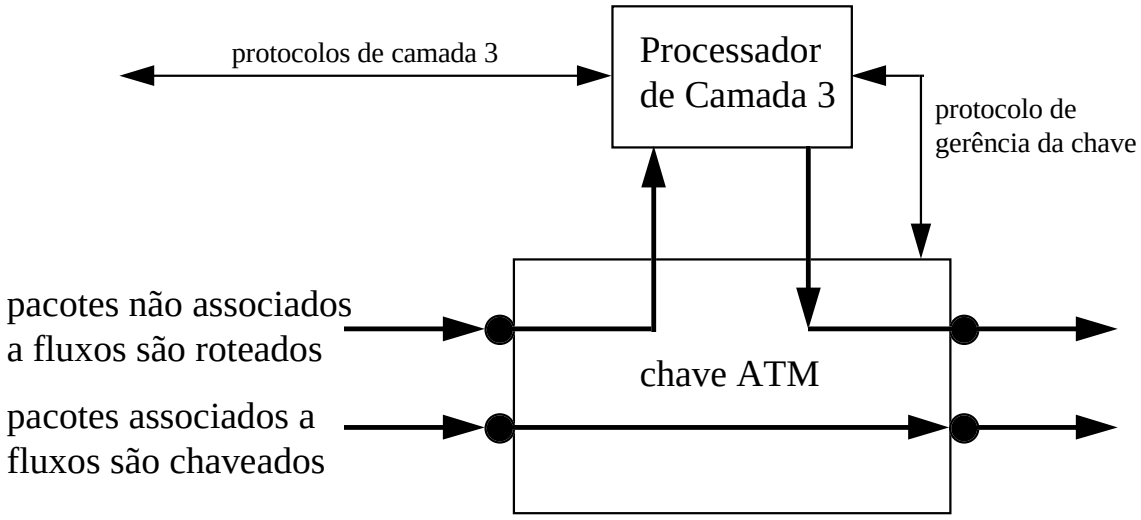


Figura 49: Chave ATM operando como *peer* de rede e explorando o conceito de fluxo.

[8] Por exemplo, células que compõem um AAL-5 CPCS-PDU transportando um datagrama IP (encapsulamento IP sobre AAL-5).

3.3 IP Switching

IP Switching é uma tecnologia desenvolvida pela Ipsilon Networks que explora o conceito de fluxo dentro do modelo *peer*. A Ipsilon já oferece produtos que incorporam esta tecnologia, assim como outros fornecedores vêm anunciando a incorporação do IP Switching em seus produtos. IP Switching elimina completamente a sinalização ATM, posicionando o roteamento exclusivamente na camada de rede (eliminando portanto a duplicidade das funções de roteamento).

A tecnologia IP Switching foi padronizada no âmbito do IETF, tornando-se portanto uma tecnologia aberta (não proprietária). A padronização consiste de três RFCs:

1. RFC 1987: Ipsilon's General Switch Management Protocol Specification Version 1.1;
2. RFC 1953: Ipsilon Flow Management Protocol Specification for IPv4 Version 1.0;
3. RFC 1954: Transmission of Flow Labelled IPv4 on ATM Data Links - Ipsilon Version 1.0.

A arquitetura IP Switching é apresentada na Figura 50. Cada chave ATM deve ser passível de gerenciamento segundo o protocolo GSMP (General Switch Management Protocol, RFC 1987), além de implementar toda a camada 3 da arquitetura TCP/IP. Via de regra a chave possui um processador a ela associado responsável pelas funções de camada 3 (denominado controlador) e pelo controle da chave via GSMP.

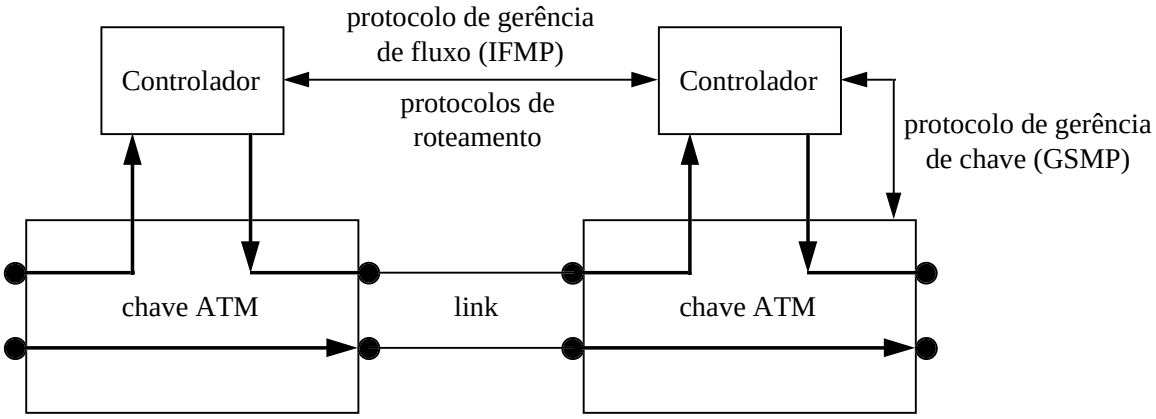


Figura 50: Arquitetura de um IP Switch.

IP Switching opera segundo o princípio da Figura 51. Datagramas IP são encapsulados no formato LLC/SNAP sobre a camada de adaptação 5 (AAL-5) e

transmitidos no canal default para este encapsulamento (VPI=0, VCI=15). O datagrama é "desencapsulado" e entregue ao controlador para roteamento convencional (caso A-1). Além de computar o roteamento, o controlador determina se o datagrama é tal que configura um fluxo. Em sendo, o controlador escolhe um par VCI/VPI disponível para a porta pela qual o datagrama foi recebido; notifica, via IFMP (Ipsilon Flow Management Protocol, RFC 1953) o nó acima (*upstream*) para a partir de então utilizar o par VPI_u/VCI_u quando transmitir datagramas deste fluxo (caso B-2). Quando estes começarem a ser recebidos pelo VCI_u/VPI_u determinado, os datagramas são encaminhados ao controlador, tal qual seus antecessores, e roteados da mesma forma (caso B-3). Quando o nó abaixo (*downstream*) detectar o mesmo fluxo e instruir este nó para utilizar o canal VPI_d/VCI_d, (casos C-4 e C-5), o controlador instrui a chave (via GSMP) para não mais encaminhar ao controlador os pacotes com VPI_u/VCI_u, mas sim chaveá-los para tal porta com VPI_d/VCI_d. A partir daí os datagramas pertencentes a este fluxo são chaveados (caso D-6), não roteados, minimizando sobremaneira o tempo de permanência do datagrama no nó.

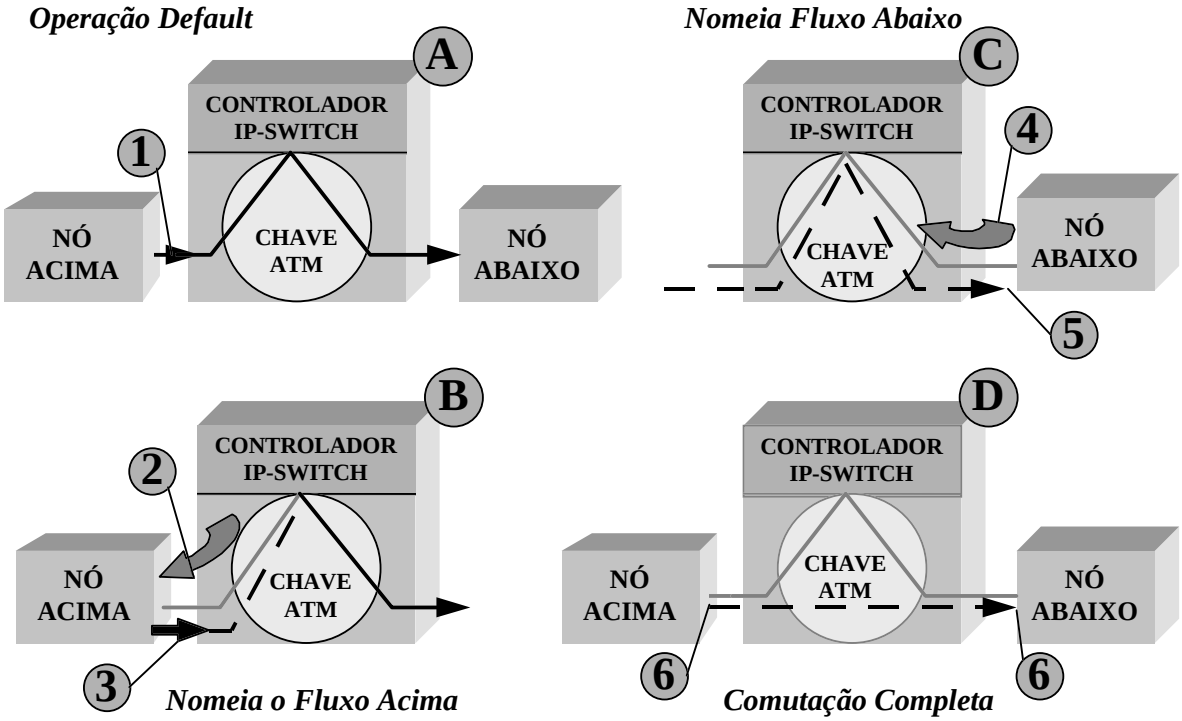


Figura 51: Operação do IP Switching.

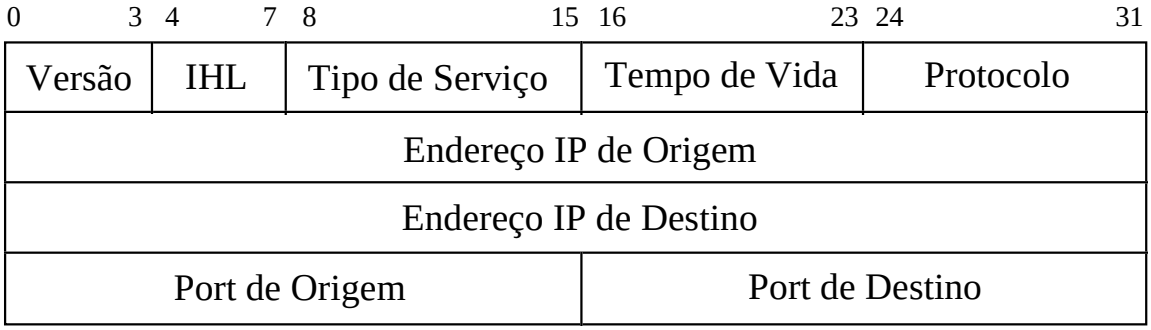
3.3.1 IFMP (Ipsilon Flow Management Protocol)

O protocolo IFMP permite que dois nós estabeleçam uma relação de adjacência e, a partir deste estabelecimento, troquem informações de redirecionamento de fluxo (associação de fluxo com VPI/VCI). Mensagens IFMP são propagadas em datagramas IP.

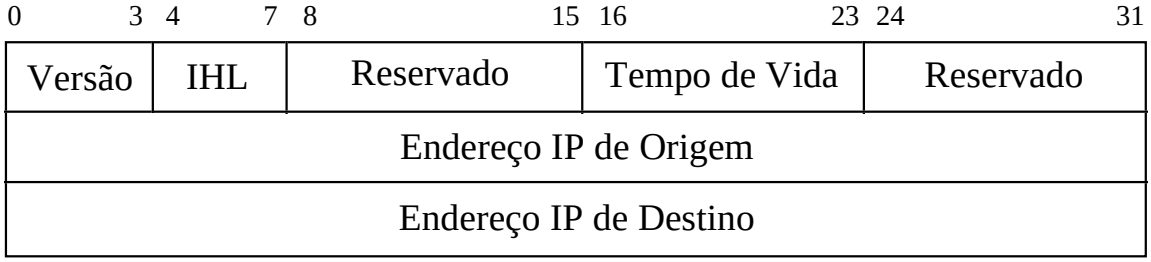
IFMP define dois tipos de fluxo[9]:

1. Fluxo tipo 1: utiliza o fluxo definido por aplicação (endereço, protocolo, port);
2. Fluxo tipo 2: utiliza o fluxo definido por destino.

A Figura 52 ilustra como os fluxos são definidos.



Fluxo tipo 1



Fluxo tipo 2

Figura 52: Tipos de fluxo definidos pelo IP Switching.

[9] A rigor existem três tipos de fluxo, sendo o fluxo tipo 0 utilizado para mudança do formato de encapsulamento de datagramas.

O campo versão deve conter a versão corrente do IFMP (versão 1). O campo IHL (Internet Header Length) informa o tamanho do descritor de fluxo em múltiplos de 4 bytes. Tipo de Serviço e Tempo de Vida são os mesmos definidos no cabeçalho do IPv4. Protocolo identifica a qual protocolo de transporte os ports estão associados (TCP ou UDP).

O protocolo de adjacência possui estrutura de mensagem ilustrada na Figura 53. A finalidade deste protocolo é manter sincronismo de estado entre dois nós conectados por um *link*.

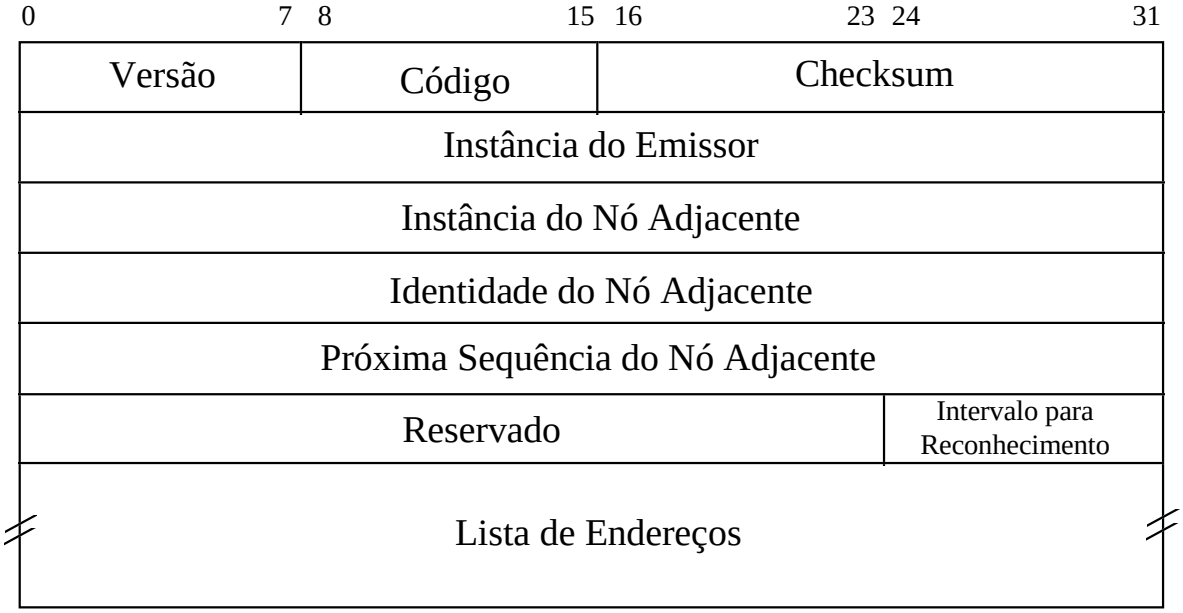


Figura 53: Estrutura de mensagem para o protocolo de adjacência do IP Switching.

Versão especifica a versão corrente do IFMP (1). Código determina o tipo de operação:

- SYN (código = 0): solicita o estabelecimento de uma relação de adjacência. É utilizada para se detectar se os nós adjacentes são capazes de processar o IFMP.
- SYNACK (código = 1): reconhece positivamente uma mensagem SYN.
- RSTACK (código = 2): termina relação de adjacência estabelecida ou em fase de estabelecimento.
- ACK (código = 3): confirma a manutenção de relação de adjacência estabelecida.

Instância do Emissor é um identificador atribuído ao nó em cada ciclo de operação. Este identificador é utilizado para que um nó adjacente interrompa uma relação de adjacência caso seu par cesse e restabeleça prontamente sua operação. Os próximos três campos se referem ao nó adjacente (*peer*):

1. Instância: o identificador de instância do nó adjacente que o nó emissor acredita ser o corrente;
2. Identidade: o endereço IP do nó adjacente que o nó emissor acredita ser o corrente;
3. Próximo Número de Sequência: indica o número da próxima mensagem IFMP que o emissor espera receber (utilizado para detecção de perdas de mensagens).

O campo Intervalo para Reconhecimento é o período que o emissor envia mensagens ACK (confirmando a manutenção da relação de adjacência).

Lista de Endereços fornece os endereços IP utilizados pelo emissor (um para cada subrede em que se conecta). Esta informação não é utilizada pelo IFMP, podendo ser útil para fins de roteamento.

Nós adjacentes (conectados por um *link*) estabelecem relação de adjacência trocando mensagens SYN e SYNACK. Esta relação é mantida pelo envio periódico de mensagens ACK. A ausência desta ou a mudança de indentificação do nó adjacente causa o *reset* do *link*, desfazendo a relação de adjacência. Nós que mantêm relação de adjacência trocam mensagens de redirecionamento de fluxo.

Mensagens de redirecionamento de fluxo (código = 4) possuem o formato dado pela Figura 54.

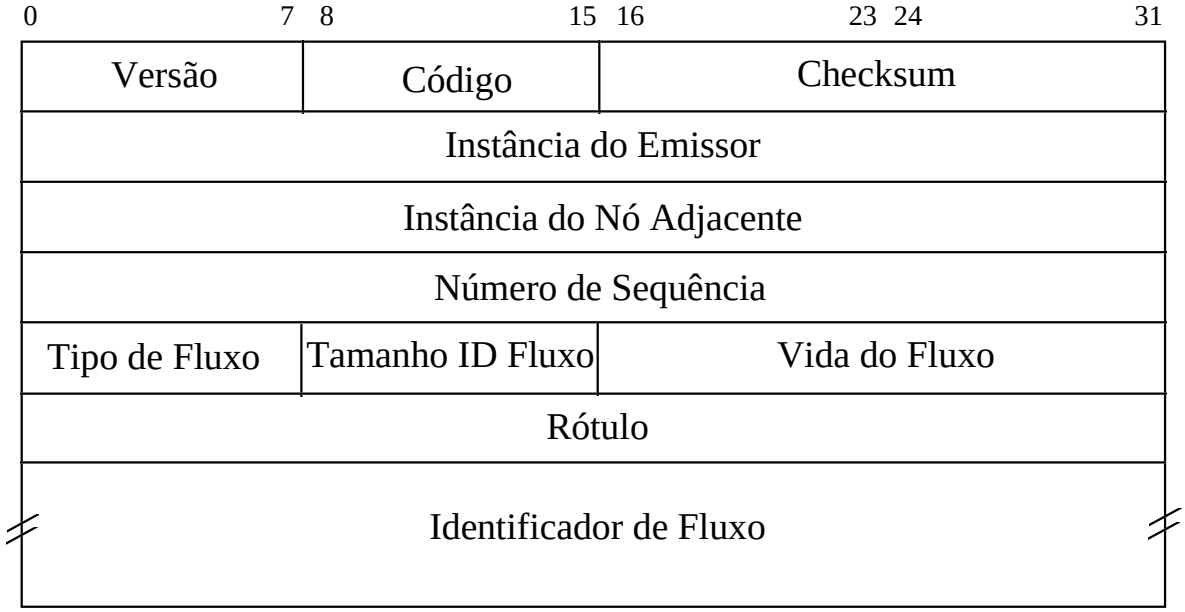


Figura 54: Mensagem de redirecionamento do IP Switching.

Os campos Tipo de Fluxo e Tamanho do Identificador de Fluxo definem o tipo de fluxo (1 ou 2). O campo Vida do Fluxo determina o número de segundos durante o qual a associação do fluxo com um par VPI/VCI é válida. O campo Rótulo identifica um fluxo segundo a RFC 1954. Esta recomendação reserva 32 bits para rotular um fluxo e estabelece que os 4 bits mais significativos são reservados, os 12 bits seguintes determinam o VPI associado ao fluxo, e os 16 bits menos significativos determinam o VCI associado ao fluxo. O campo Identificador de Fluxo contém os campos necessários para identificar um fluxo tipo 1 ou 2.

Redirecionamento de fluxo faz com que datagramas pertencentes a um determinado fluxo sejam transmitidos por um canal específico, diferente do canal default (VPI=0, VCI=15). O encapsulamento de datagramas redirecionados segue a recomendação da RFC 1954 e difere do encapsulamento padrão LLC/SNAP. Em linhas gerais, o encapsulamento de datagramas redirecionados é feito diretamente sobre o CPCS-PDU da camada de adaptação número 5 (sem o cabeçalho LLC/SNAP) e removendo-se os campos do datagrama IP presentes no descritor de fluxo (Figura 54).

Em oposição ao redirecionamento de fluxo, a mensagem de desassociação (código = 5) faz com que os datagramas pertencentes a um determinado fluxo sejam desassociados de um determinado canal virtual (par VPI/VCI) e tornem a ser transmitidos pelo canal default. Uma mensagem de reconhecimento para a mensagem de desassociação é definida (código = 6). A mensagem possui o mesmo formato da mensagem de redirecionamento (Figura 54), exceto o campo Vida do Fluxo possui agora um valor reservado.

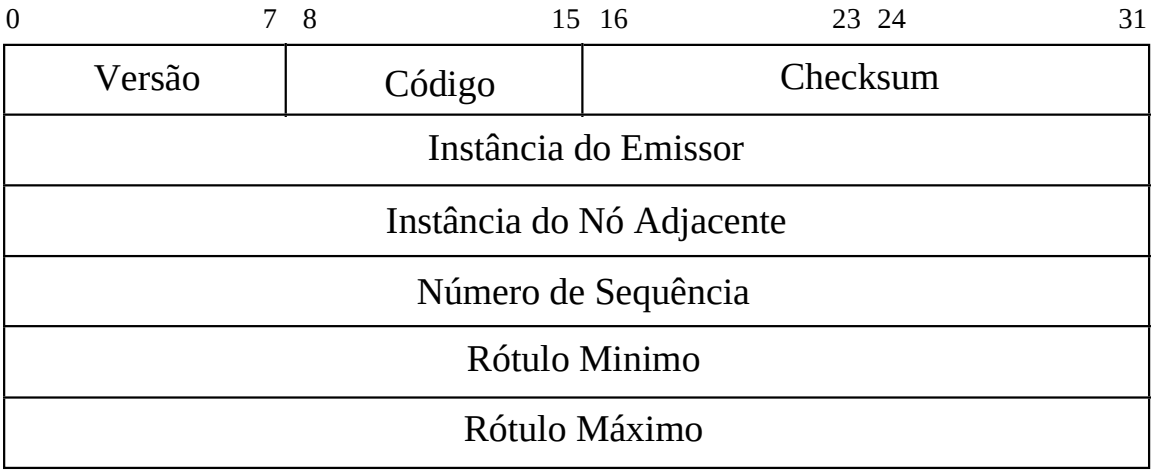
O protocolo IFMP provê ainda mecanismos para um nó informar o intervalo para rótulos que é capaz de manipular. Esta informação deve ser levada em conta quando um nó abaixo (*downstream*) associa um fluxo a um determinado canal virtual. Mensagens tipo intervalo de rótulo (código = 7) são utilizadas para esta finalidade. Finalmente, o protocolo IFMP possui um tipo de mensagem para o informe de erros (código = 8). Dois códigos de erros são definidos: versão de protocolo e tipo de fluxo incorretos (não suportados). Um campo adicional provê informação complementar ao erro: respectivamente, a maior versão suportada e tipo de fluxo que causou o erro. A Figura 55 ilustra o formato destas mensagens.

3.3.2 GSMP (Ipsilon's General Switch Management Protocol)

O GSMP é um protocolo de gerência de chaves ATM. O protocolo emprega mensagens tipo requisição-resposta entre um controlador (no caso, o processador de camada 3) e a chave ATM. O GSMP define seis tipos de operações:

1. gerência de conexões;
2. gerência de portas;
3. obtenção de estatísticas de operação da chave;

4. configuração da chave e de portas;
5. estabelecimento de relação de adjacência;
6. notificação de eventos.



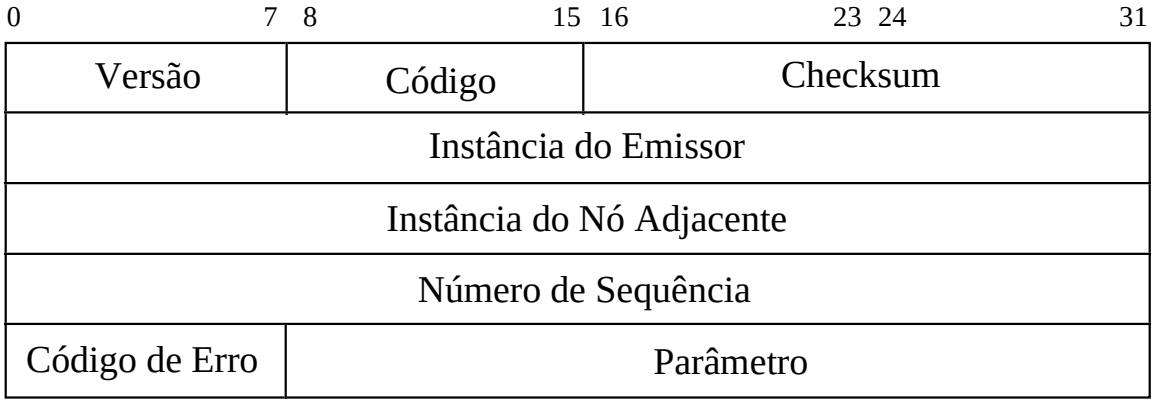


Figura 55: Formatos de mensagens para intervalo de rótulos (acima) e informe de erros (abaixo).

A Figura 56 ilustra uma chave sendo controlada via GSMP. Mensagens GSMP são encapsuladas no formato LLC/SNAP e encaminhadas no canal VPI=0, VCI=15 (mesmo canal para datagramas ainda não associados a fluxos).

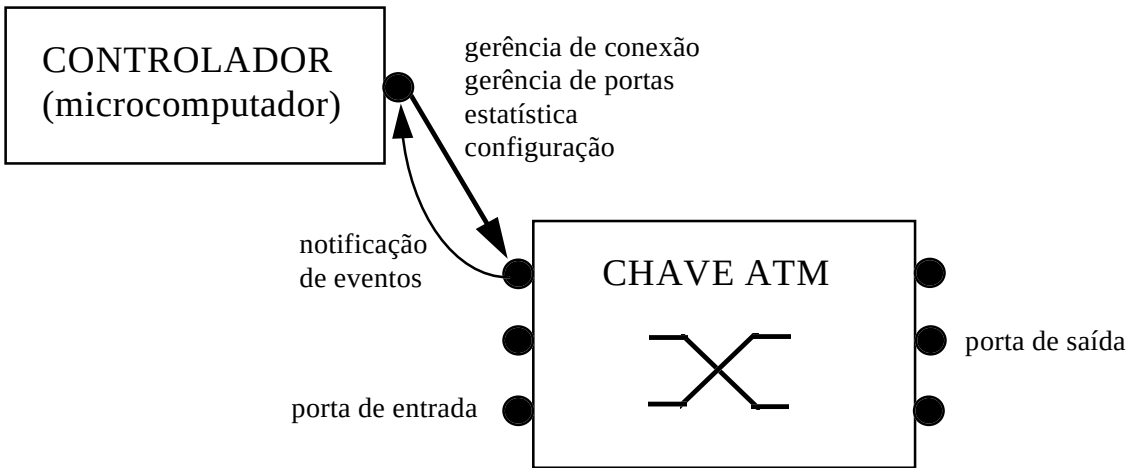


Figura 56: Chave ATM controlada via GSMP.

O protocolos GSMP possui formato de mensagens segundo a Figura 57.

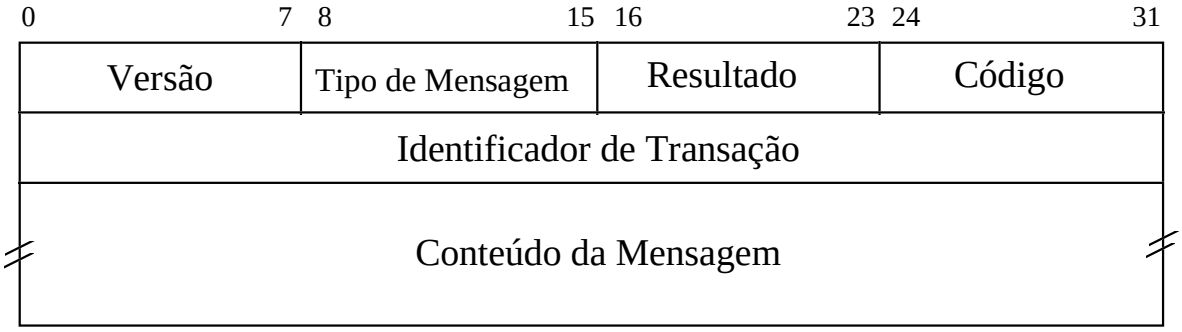


Figura 57: Formato de mensagem GSMP.

O campo Versão informa a versão corrente do protocolo GSMP (1). O campo Tipo de Mensagem determina o tipo de operação, dentre os seis enumerados anteriormente. O campo Resultado informa se:

- nenhum resultado deve ser retornado em caso de sucesso (Resultado = 1);
- resultado requerido em caso tanto de sucesso quanto de falha (Resultado = 2);
- informe apenas em caso de sucesso (Resultado = 3);
- informe apenas em caso de falha (Resultado = 4).

O campo Código provê uma informação complementar ao campo Resultado (o motivo da falha, por exemplo). Finalmente, dado que o emissor da requisição não bloqueia aguardando a resposta, o campo Identificador de Transação é necessário para associar uma resposta a sua respectiva requisição.

Gerência de Conexões

Gerência de conexões é empregada para estabelecer, remover, modificar e verificar o estado de conexões através da chave. Uma conexão é identificada através do número da porta de entrada e do par VPI/VCI associado à conexão nesta porta. No GSMP toda a conexão é ponto-multiponto (p-mp), sendo a conexão ponto-ponto (p-p) um caso particular desta. Uma ramificação de conexão ponto-multiponto é identificada pela porta de saída e pelo par VPI/VCI correspondente à ramificação nesta porta.

A Figura 5 ilustra o formato de uma mensagem GSMP de gerência de conexões.

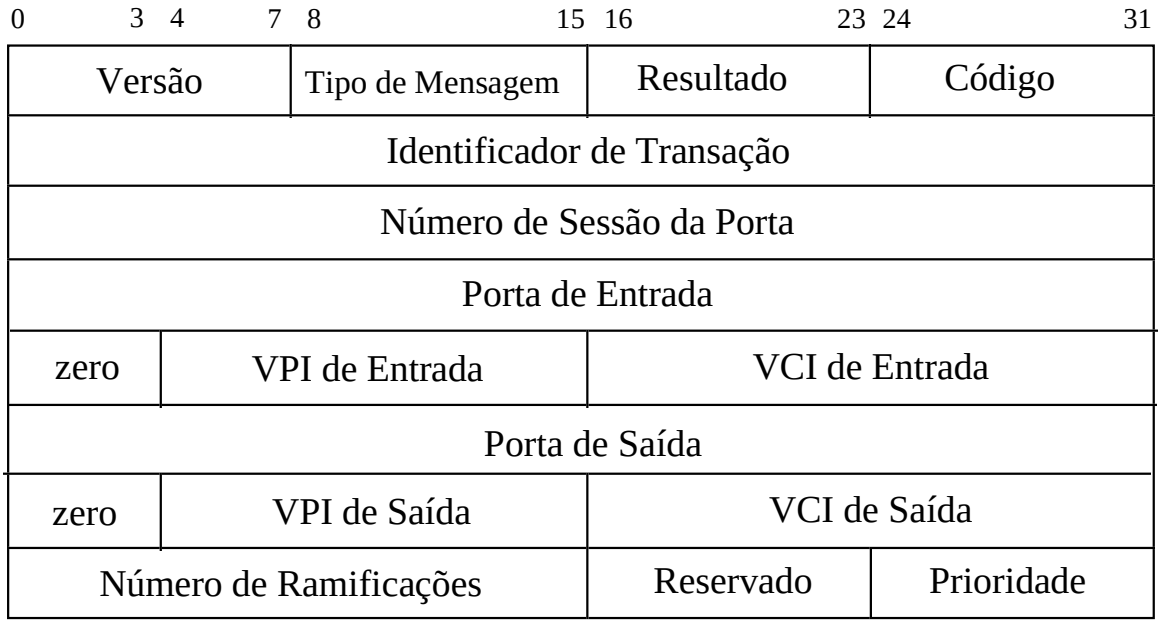


Figura 58: Mensagem GSMP de gerência de conexões.

Uma porta possui um identificador próprio (Porta de Entrada/Saída) e um identificador de sessão (Número de Sessão da Porta) , atualizado toda a vez que a porta entrar no estado de disponibilidade. O campo Número de Ramificações identifica o número de ramificações de uma conexão (1 para conexões p-p, 2 ou mais para conexões p-mp).

O campo Prioridade determina a prioridade da conexão em relação às demais estabelecidas através da chave. Este é o único recurso que o protocolo GSMP possui para o tratamento de qualidade de serviço. É responsabilidade da chave incorporar prioridades em sua política de controle de admissão e de tráfego.

O protocolo define 6 operações de gerência de conexões, segundo o valor constante no campo Tipo de Mensagem:

- Adição de uma ramificação a uma conexão (Tipo = 16). Se a conexão inexiste é criada uma conexão ponto-ponto. Caso exista nesta forma, a mesma é transformada em ponto-multiponto com duas ramificações. Caso a conexão já seja ponto-multiponto, uma nova ramificação é adicionada.

- Remoção de uma ramificação a uma conexão (Tipo = 17). Caso a conexão seja do tipo ponto-multiponto com duas ramificação, após a operação a mesma se tranforma em ponto-ponto. Caso a conexão seja ponto-ponto, esta operação causa a remoção da conexão.

- Remoção da árvore de conexão (Tipo = 18). A conexão indentificada por (porta, VPI/VCI) é removida inteiramente.

- Inspeção da árvore de conexões (Tipo = 19). Esta operação retorna o número de ramificações da conexão.

- Remoção de todas as conexões associadas à determinada porta (Tipo = 20). Remove todas as conexões cuja identificação possui a porta dada no campo Porta de Entrada.

- Alteração em ramificação (Tipo = 22). Associa um novo valor (porta, VPI/VCI) a uma dada ramificação.

Gerência de Portas

Gerência de portas permite controlar uma determinada porta da chave. A Figura 59 ilustra o formato de mensagens GSMP para gerência de portas. O campo Tipo de Mensagem possui o valor 32 para esta operação. A porta a ser gerenciada é identificada pelo campo Identificador da Porta e Número de Sessão da Porta. O campo Número de Sequência de Eventos informa ao controlador o valor corrente do número de sequência na geração de eventos associados à porta em questão (este campo é utilizado apenas em mensagens de resposta). Esta informação permite ao controlador detectar se algum evento gerado foi perdido.

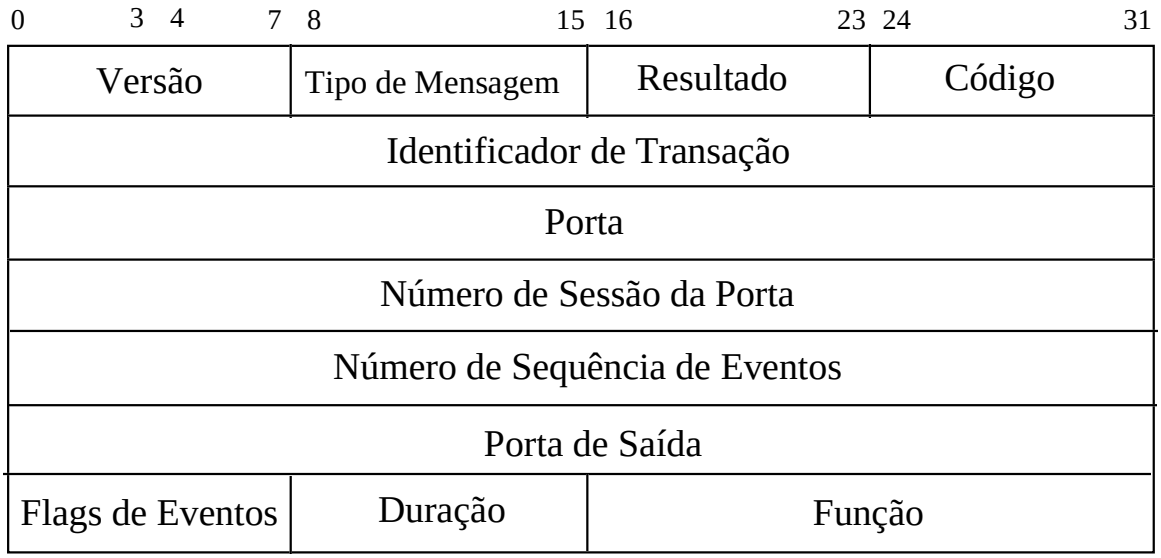


Figura 59: Mensagem GSMP de gerência de portas.

O campo Flags de Eventos é utilizado para *reset* de eventos (mensagens de requisição) ou informe do flag corrente de eventos associado à porta (mensagens de resposta). Os eventos associados a uma porta são:

- porta em operação;
- porta fora de operação;
- VCI/VPI inválido;
- nova porta em operação;
- porta fora de operação.

O campo Duração determina o tempo em que a porta deve permanecer em estado de *loopback*, caso a operação coloque a porta neste modo. O campo Função determina a operação sobre a porta:

- coloca a porta em operação;
- retira a porta de operação;
- coloca a porta em *loopback* (“curto-circuita”a porta);
- “resseta” a porta (removendo todas as conexões associadas à porta);
- “resseta” o flag de eventos.

Obtenção de Estatísticas

O protocolo GSMP permite inquirir a chave sobre a valor de contadores mantidos pelo hardware. Estes contadores são associados com as atividades que ocorrem nas

portas de entrada e saída e, eventualmente, nas conexões. O campo Tipo de Mensagem do GSMP possui os seguintes valores:

- 48: solicita o tráfego (bytes/s) associado à determinada conexão;
- 49: solicita estatísticas associadas a determinada porta;
- 50: solicita estatísticas associadas à detrminada conexão (porta, VPI/VCI).

Os valores constantes numa mensagem de resposta para mensagens tipo 49 e 50 referenciando uma porta de entrada são:

- número de células recebidas;
- número de quadros recebidos;
- número de quadros descartados;
- número de células descartadas por *checksum* (estatísticas por porta apenas);
- número de células com VPI/VCI inválido (estatísticas por porta apenas).

Os valores constantes numa mensagem de resposta para mensagens tipo 49 e 50 referenciando uma porta de saída são:

- número de células transmitidas;
- número de quadros transmitidos;
- número de células descartadas;
- número de quadros descartados.

Configuração da Chave e de Portas

Este tipo de operação permite ao controlador inquirir a chave, uma porta, ou todas as portas sobre sua configuração. Mensagens de configuração da chave (campo Tipo de Mensagem igual a 64) permite ao controlador descobrir:

- a versão do firmware da chave;
- o tipo da chave;
- o nome da chave (tipicamente seu endereço MAC no formato IEEE 802).

Mensagens de configuração de portas (campo Tipo de Mensagem igual a 65) permite ao controlador descobrir:

- os valores máximos e mínimos de VPI e VCI que podem ser associados à porta;
- a banda máxima para a porta (células por segundo);
- o estado da porta (operacional, não operacional ou *loopback*);
- o tipo da porta (SONET/SDH, DS-3, etc.) segundo a RFC 1573;
- o estado do link conectado à porta (ativo, inativo, em teste);
- a faixa de prioridades para as conexões asssociadas a porta.

Caso o campo Tipo da Mensagem contenha o valor 66, os dados acima são fornecidos ao controlador para cada porta da chave.

Estabelecimento de Relação de Adjacência

O GSMP define um protocolo de *handshake* entre o controlador e chave idêntido ao utilizado pelo IFMP. A finalidade é manter sincronização de estados entre controlador e chave.

Notificação de Eventos

Notificação de eventos é uma atividade assíncrona, isto é, ocorre sem que o controlador requisite. Ao receber uma notificação de evento o controlador age em função do tipo de evento sendo reportado. Nenhuma notificação de recepção de evento é fornecida à chave por parte do controlador. Eventos são associados a portas e os tipos de eventos foram listados no item Gerência de Portas.

3.3.3 IP Switching: Vantagens e Desvantagens

A exploração do conceito de fluxo é importante hoje no âmbito da Internet, a mais importante rede de comunicação de dados do mundo. Nesta rede, o World Wide Web e as aplicações multimídia geram longas sequências de pacotes de uma fonte para um único ou múltiplos destinos (isto é, geram fluxo). Portanto, a vantagem primordial de tecnologias como o IP Switching é a própria exploração do conceito de fluxo visando minimizar o esforço de roteamento. Outra vantagem do IP Switching é sua disponibilidade: tanto a Ipsilon, criadora da tecnologia, quanto outros fabricantes (Digital, por exemplo) anunciaram a incorporação da tecnologia em seus produtos ATM. Ainda outra vantagem é a eliminação da duplicidade de funções de roteamento (IP Switching não utiliza a sinalização ATM).

Entretanto, algumas desvantagens devem ser consideradas:

- alocar uma conexão por fluxo pode exaurir a capacidade de conexões das chaves, principalmente em WANs e grandes *backbones*;
- uma vez detectado um fluxo, sua duração é imprevisível (portanto, o desempenho da tecnologia depende do perfil das aplicações que fazem uso da rede);
- todas as chaves e roteadores de uma determinada malha devem dispor desta tecnologia para tirar proveito do chaveamento de pacotes na camada ATM;
- a aceitação plena e em larga escala desta tecnologia ainda é um fator de incerteza.

3.4 Tag Switching

Tag Switching é uma proposta da Cisco Systems na linha de explorar o conceito de fluxo e integrar, sem duplicações, funcões de camada 3 com comutação ATM. Em sendo uma tecnologia associada ao modelo *peer*, Tag Switching exige que cada elemento de rede (nó) implemente certas funcionalidades da camada de rede do modelo OSI.

A proposta se encontra no estágio de "Internet Draft" (ID), isto é, pré-RFC. Três IDs descrevem a tecnologia:

1. Rekhter, Y. et. al., Tag Switching Architecture - Overview (draft-rekhter-tagswitch-arch-00.txt), Janeiro de 1997;
2. Doolan, P., et. al., Tag Distribution Protocol (draft-doolan-tdp-spec-01.txt), Maio 1997;
3. Davie, B., et. al., Use of Tag Switching With ATM (draft-davie-tag-switching-atm-01.txt), Janeiro de 1997.

Uma diferença fundamental entre as tecnologias IP Switching e Tag Switching é que a primeira define fluxo baseado no tráfego de dados, enquanto a segunda define fluxo baseado no tráfego de controle (informações de roteamento). Outra diferença importante é que Tag Switching não está diretamente associada com tecnologia ATM[10], podendo, por exemplo, ser integrada a um roteador convencional.

Tag Switching permite definir fluxo de várias maneiras e com diferentes granularidades, por exemplo:

- pacotes associados a determinado prefixo (subrede) de destino (independente da origem);
- pacotes com rota pré-fixada (roteados na origem);
- pacotes que devem ser propagados por uma árvore de multicast;
- pacotes que trafegam entre dois hosts ou duas aplicações (tal qual IP Switching).

Inicialmente a tecnologia irá privilegiar a associação de fluxo com rotas, atribuindo rótulos (tags) a destinos[11]. Esta associação é processada em determinado nó e propagada para os nós vizinhos através de um "protocolo de distribuição de tags" (TDP: Tag Distribution Protocol). A idéia básica é utilizar os tags para aumentar a eficiência do roteamento.

A Figura 60 ilustra um arranjo típico para o emprego de Tag Switching. O arranjo é similar ao da Figura 48, exceto que os roteadores de borda (Tag Edge Routers) e as chaves interiores à nuvem ATM (os Tag Switch Routers) são capazes de operar os

[10] Apesar de ser fortemente motivada por esta tecnologia.
[11] Este texto irá se limitar a este caso.

protocolos convencionais de roteamento (OSPF, BGP, etc.) e os protocolos a serem definidos para Tag Switching (TDP, etc.). Um roteador de borda ao receber um pacote para encaminhamento:

1. analisa o cabeçalho do protocolo da camada de rede (cabeçalho IP, por exemplo);
2. realiza os serviços de rede aplicáveis (segurança, roteamento, etc.);
3. seleciona uma rota para o pacote baseado em sua tabela de roteamento;
4. aplica o tag associado à rota e encaminha o pacote para o próximo nó (um Tag Switch);
5. o Tag Switch analisa o pacote e comuta tendo como base somente o tag sem analisar o cabeçalho do protocolo da camada de rede;
6. o pacote alcança o roteador de borda de saída, o qual extrai o tag e entrega o pacote (ou o encaminha à outra subrede na rota, eventualmente com um novo tag).

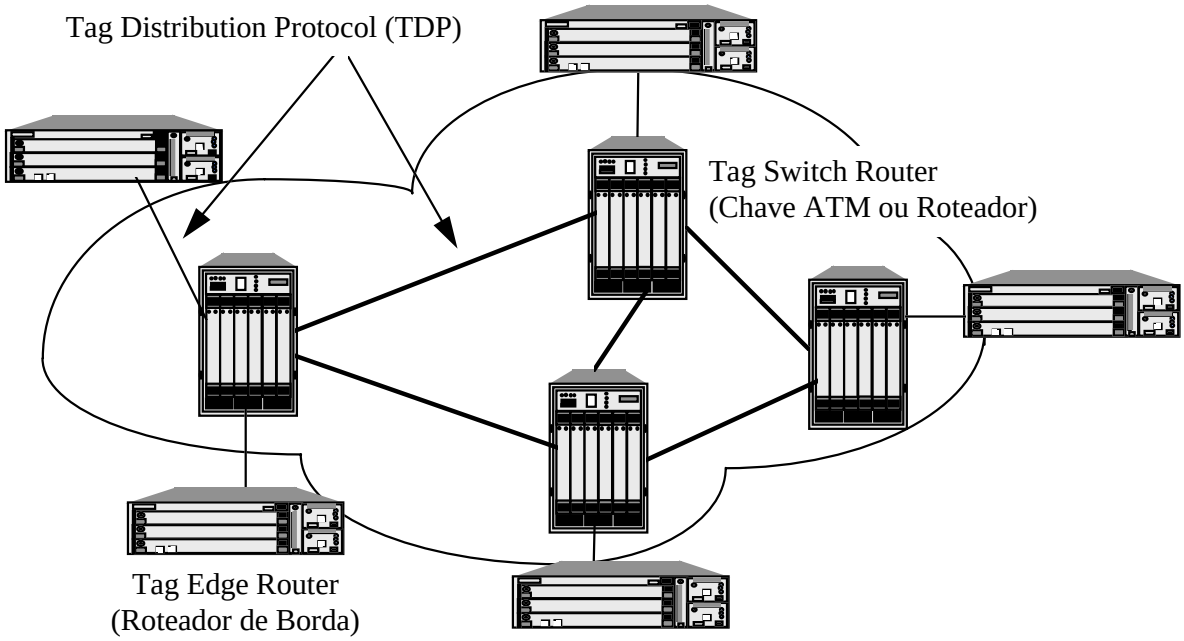


Figura 60: Cenário de operação do TAG Switching.

A tecnologia Tag Switching prevê dois componentes básicos:

1. Componente de Encaminhamento;
2. Componente de Controle.

O equipamento de rede que possui estes dois componentes é denominado genericamente de TSR (Tag Switching Router). O componente de encaminhamento executa o encaminhamento de pacotes baseado no tag associado ao destino do

pacote. O componente de controle é responsável por manter a associação tag-destino e distribuir esta associação entre chaves conectadas. Esta distribuição pode ocorrer na "carona" de algum protocolo de roteamento ou através do protocolo TDP.

3.4.1 Componente de Encaminhamento

TSRs mantém um mapeamento tag-destino numa Base de Informação de Tag (TIB: Tag Information Base). Um TSR pode manter uma única TIB, uma TIB por interface ou uma TIB por grupo de interfaces. A TIB é uma tabela onde cada entrada possui o formato abaixo:

TAG de Entrada	TAG de Saída	Interface de Saída	Informação de Camada 2	Prefixo de destino

Recebido um pacote acompanhado de um tag o TSR consulta a tabela procurando casar este tag com o campo TAG de Entrada. Caso o tag não exista na TIB, o TSR pode rotear o pacote de maneira convencional, ou até descartá-lo. Caso exista:

1. o tag é trocado pelo tag de saída[12];
2. o pacote com o novo tag é propagado via interface definida no campo Interface de Saída;

O campo Informação de Camada 2 armazena informações necessárias para montar um quadro enlace, por exemplo, o endereço MAC do próximo *hop* (nó) da rota.

Os campos TAG de Saída, Interface de Saída e Informações de Camada 2 de uma dada entrada podem se repetir no caso do pacote se endereçar múltiplos destinos (multicast) e o TSR se constituir de uma ramificação na árvore de multicast.

Tag Switching permite propagar tags em pacotes de três maneiras:

1. entre os cabeçalhos do quadro de enlace e o pacote de rede;
2. em algum campo do quadro de enlace (exemplo: VPI/VCI de células ATM);
3. em algum campo do pacote de rede (exemplo: camplo Identificador de Fluxo do IPv6).

Note que nenhuma destas formas é passível de implementação imediata sem ação de padronização.

[12] Esta técnica é denominada genericamente *label swapping*.

3.4.2 Componente de Controle

O componente de controle associa tags com rotas (destinos) e distribui esta associação. Da mesma forma que as tabelas de roteamento[13] são construídas através da interação entre roteadores via protocolos de roteamento, TIBs são construídas através da interação entre TSRs via protocolo de distribuição de tags (TDP).

Existem três maneiras de se alocar e distribuir tags:

1. alocação pelo TSR abaixo (*downstream*);
2. alocação pelo TSR abaixo sob demanda;
3. alocação pelo TSR acima (*upstream*).

A relação entre TSRs (abaixo ou acima) é em relação ao sentido do fluxo de pacotes. Na alocação *downstream* o TSR cria associações tag-destino para os pacotes que chegam às suas interfaces e distribui estas associações para os TSRs conectados às suas interfaces (TSRs acima). Analogamente, este TSR recebe associações atribuidas por TSRs abaixo. A alocação *downstream* por demanda é idêntica, exceto que o TSR abaixo gera associações somente se solicitado pelo TSR acima. Alocação *upstream* é dual da alocação *downstream*: o TSR cria associações para os pacotes que saem de suas interfaces e recebe associações para os pacotes que chegam às suas interfaces. Aparentemente, o método a ser utilizado nas implementações é a alocação *downstream* por demanda por gerar menor tráfego na distribuição das associações. A Figura 61 ilustra as alocações *downstream* e *upstream*.

[13] Denominadas FIB (Forward Information Base).

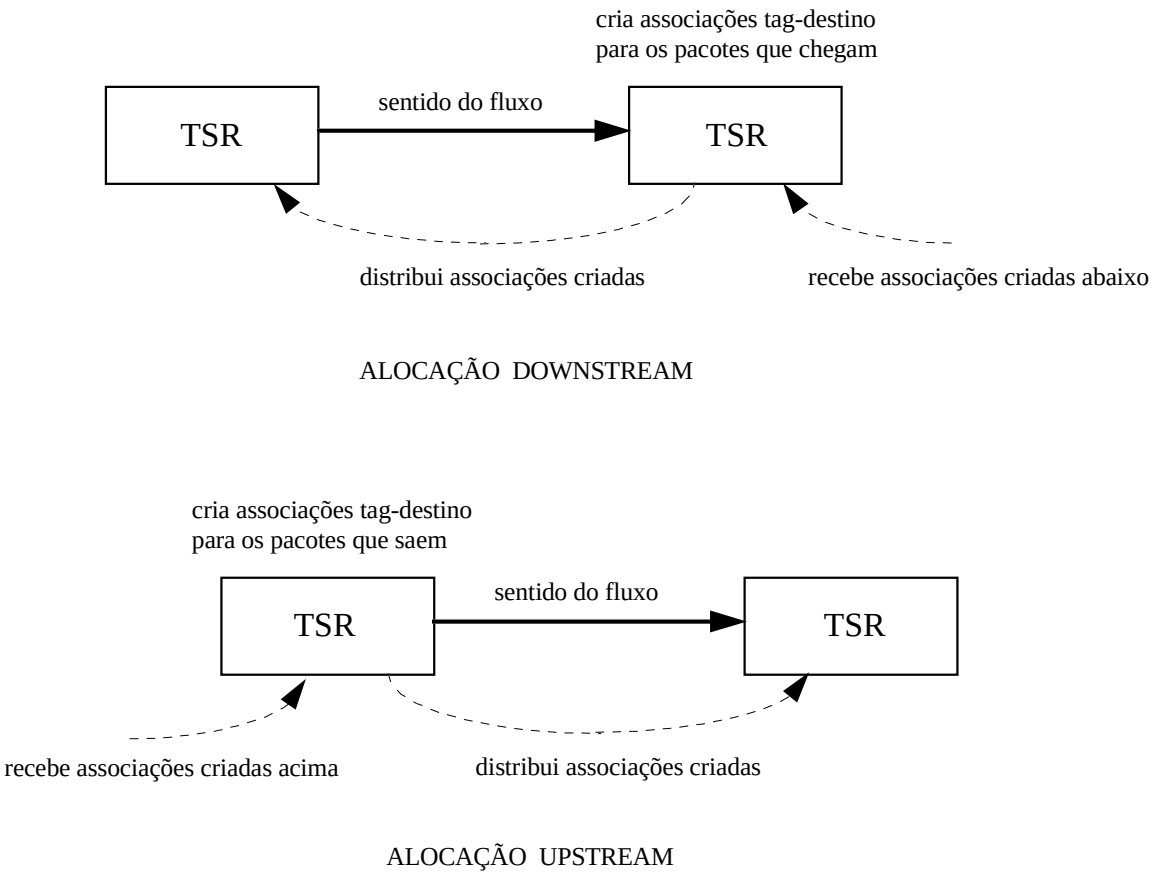


Figura 61: Alocações *downstream* e *upstream***.**

Seja o caso de alocação *downstream*. Para cada entrada na tabela de roteamento (FIB) o TSR cria um novo tag e associa ao destino correspondente. Esta associação é armazenada na TIB da seguinte maneira (ver Figura 62):

- o tag criado é armazenado no campo TAG de Entrada;
- o campo TAG de Saída é deixado em branco;
- o campo Interface de Saída é o mesmo dado pela tabela de roteamento para o próximo *hop* da rota;
- o campo Informações de Camada 2 é preenchido (se for o caso) com informações da interface.

Ao receber do TSR abaixo uma associação, o TSR verifica se sua TIB possui entrada cujo destino coincide com o campo Prefixo de Destino. Caso possua, o tag recebido é utilizado para completar o campo TAG de Saída. A Figura 62 ilustra este procedimento.

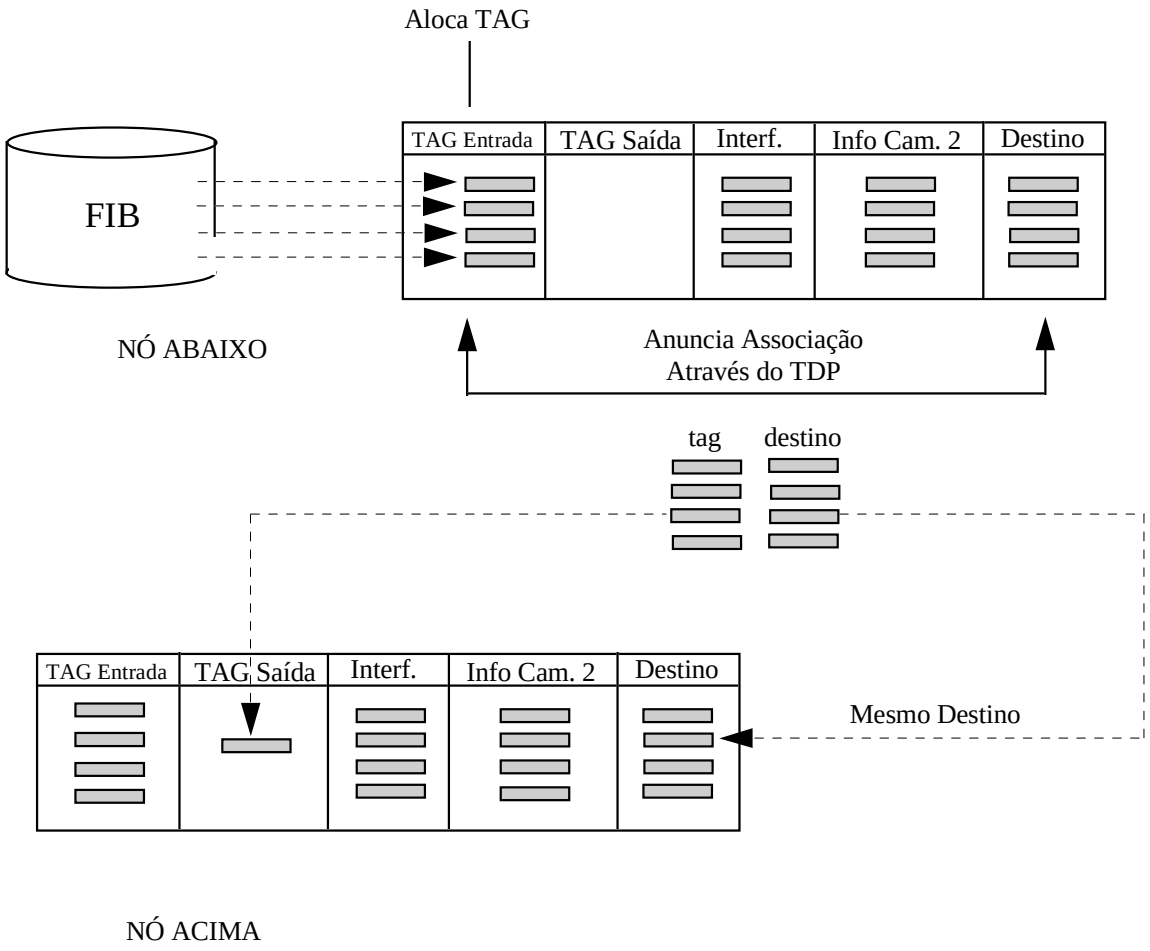


Figura 62: Criação de associações tag-destino.

Quando todas as TIBs forem completadas, ao receber um pacote de um TSR com um tag, o receptor inspeciona sua TIB a procura do tag no campo TAG de Entrada. Encontrada a entrada, o receptor troca o tag (campo TAG de Saída) e propaga o pacote na interface correspondente. Nenhum processamento de camada 3 adicional é necessário. Este procedimento, segundo a Cisco, mesmo operando numa rede baseada unicamente em roteadores convencionais aumenta o desempenho da rede pois a troca de tags é muito mais rápida que o roteamento completo de camada 3, o que faz diminuir o atraso na propagação do pacote. Vale observar que um nó equipado com os protocolos Tag Switching necessita ainda da capacidade plena de rotear pacotes. Em outras palavas, todos os TSRs devem possuir capacidades plenas de roteadores.

Roteamento Hierárquico

A tecnologia Tag Switching permite também desacoplar o roteamento interior do roteamento exterior. Seja o topologia da Figura 63 constituindo um domínio sem operar Tag Switching. Os roteadores de borda executam protocolos de roteamento exterior, por exemplo, BGP (Border Gateway Protocol). Todos os roteadores do domínio executam um protocolo de roteamento interior (por exemplo, OSPF). Se o domínio é utilizado como "passagem" para outros domínios[14], os roteadores interiores necessitam também armazenar rotas exteriores ao domínio (para poderem escolher o melhor roteador de saída). Isto implica em grandes tabelas de roteamento com os problemas de ineficiência e longo tempo de convergência para um consenso quanto a topologia da rede.

Tag Switching permite um desacoplamento do roteamento interior e exterior, isto é, TSRs interiores armazenariam informação apenas de roteamento interior. Para tal, um pacote em trânsito pelo dominio carrega duas tags: uma utilizada para rotas internas e outra para rotas externas. O pacote chega a um roteador de borda carregando um único tag associado ao seu destino (Figura 63). Este roteador adiciona um segundo tag utilizado para dirigir o pacote ao roteador de borda de saída. Este, por sua vez, remove o segundo tag, troca o tag original para aquele definido pelo roteador de entrada do próximo domínio, e propaga o pacote para fora do domínio.

Para viabilizar este processo, os roteadores de borda mantém tags associadas à destinos externos (outros domínios) e tags associadas a destinos internos (outros roteadores de borda). Roteadores de borda propagam para o interior do domínio associações com os destinos externos atingíveis a partir deste roteador. Os demais roteadores de borda, recebendo esta informação, associam dois tags ao destino externo: o tag propagado pelo roteador de borda e o tag referente ao destino que representa o próprio roteador de borda que propagou a associação. Este último tag foi obtido via distribuição de tags associados à destinos interiores ao domínio (isto é, construído com base nas tabelas de roteamento interior).

TSRs interiores não associam tags a destinos externos.

[14] Este é o caso de grandes provedores de acesso à Internet.

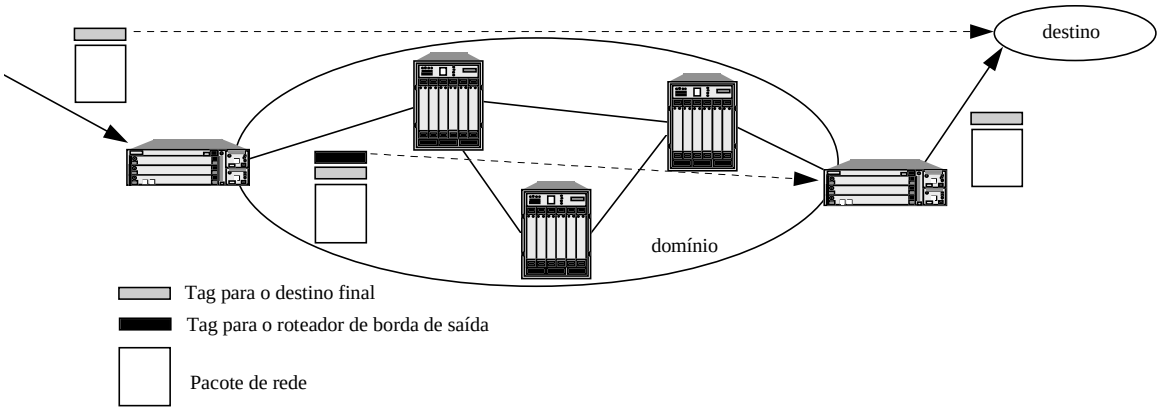


Figura 63: Desacoplamento entre roteamento interior e exterior.

Qualidade de Serviço (QoS)

Tag Switching pode prover qualidade de serviço associando um tag a uma classe de serviço ou a uma sessão estabelecida via protocolo RSVP (Resource Reservation Protocol). No primeiro caso a classe de serviço deve estar presente no tag, por exemplo, nos três bits de maior ordem. Tag Chaves tratam o pacote de acordo com a classe de serviço a ele associada.

No segundo caso, uma pequena extensão do RSVP se faz necessária para propagar um tag durante o procedimento de reserva de recursos. Pacotes com este tag são associados com uma determinada sessão RSVP e recebem tratamento adequado à sua sessão. Note que este caso é similar ao processo *downstream* de alocação de tags, exceto que agora tags são distribuídos via RSVP, não TDP.

3.4.3 Tag Distribution Protocol (TDP)

TDP permite a um Tag Switch Router (TSR):

- distribuir associações tag-destino;
- requisitar associações;
- liberar (invalidar) associações;

TDP opera sempre entre dois TSRs conectados e exige um protocolo de transporte confiável (por exemplo TCP) ou encapsulamento LLC/SNAP caso os TSRs conectados sejam chaves ATM . Um TSR pode manter sessões TDP com múltiplos TSRs conectados. Cada sessão é identificada por um “Identificador TDP de Sessão”.

Quando um TSR detecta a perda de uma sessão (por expiração de timer, perda de conexão de transporte, etc.), o mesmo destroi todas as associações recebidas do TSR com o qual mantinha a sessão.

TDP possui um cabeçalho dado pela Figura 64.

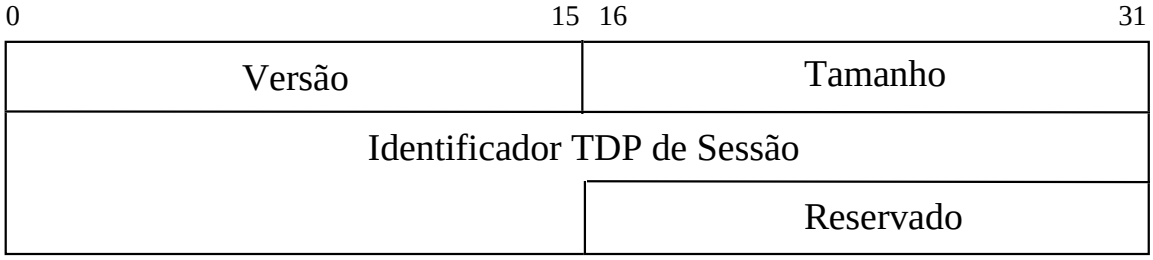


Figura 64: Cabeçalho do protocolo TDP.

O campo Versão deve conter a versão atual do protocolo (1). O campo Tamanho contém a tamanho do PDU em bytes a partir deste campo (isto é, o tamanho total do PDU menos 4). O campo Identificador TDP de Sessão é utilizado para identificar uma sessão TDP. Dos 6 bytes deste campo, os quatro iniciais contém o endereco IP do TSR e os dois finais representa a sessão TDP mantida pelo TSR.

O campo de dados do protocolo carrega PIEs (Protocol Information Elements). Um PIE possui a forma tipo-tamanho-valor, conforme ilustrado na Figura 65.

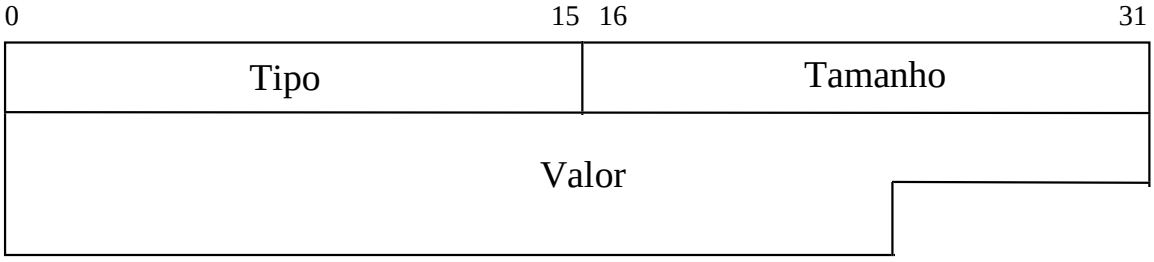


Figura 65: Formato de um PIE (Protocol Information Element).

Existem sete tipos de PIEs:

1. OPEN: utilizado para iniciar uma sessão TDP;
2. KEEPALIVE: utilizado para confirmar um OPEN;
3. NOTIFICATION: utilizado para informe de erros e encerrar uma sessão;
4. BIND: utilizado para propagar associações;

5. REQUEST_BIND: utilizado para requisitar associações (alocação *downstream* sob demanda);
6. WITHDRAW_BIND: utilizado para instruir o receptor para descartar imediatamente determinada associação;
7. RELEASE_BIND: utilizado para cancelar uma requisição tipo REQUEST_BIND.

Na versão atual do TDP um tag possui 32 bits. As associações podem ser ser de sete tipos (BLIST_TYPE):

- tipo 0: associação nula (tag apenas, ulitizada por exemplo no WITHDRAW_BIND);
- tipo 1: alocação *upstream*;
- tipo 2: alocação *downstream*;
- tipo 3: alocação *upstream* com destino multicast e origem não especificada;
- tipo 4: alocação *upstream* com destino multicast e origem especificada;
- tipo 5: alocação *upstream* onde o tag é um VCI;
- tipo 6: alocação *downstream* onde o tag é um VCI.

A Figura 66 ilustra uma mensagem BIND, utilizada para propagar associações.

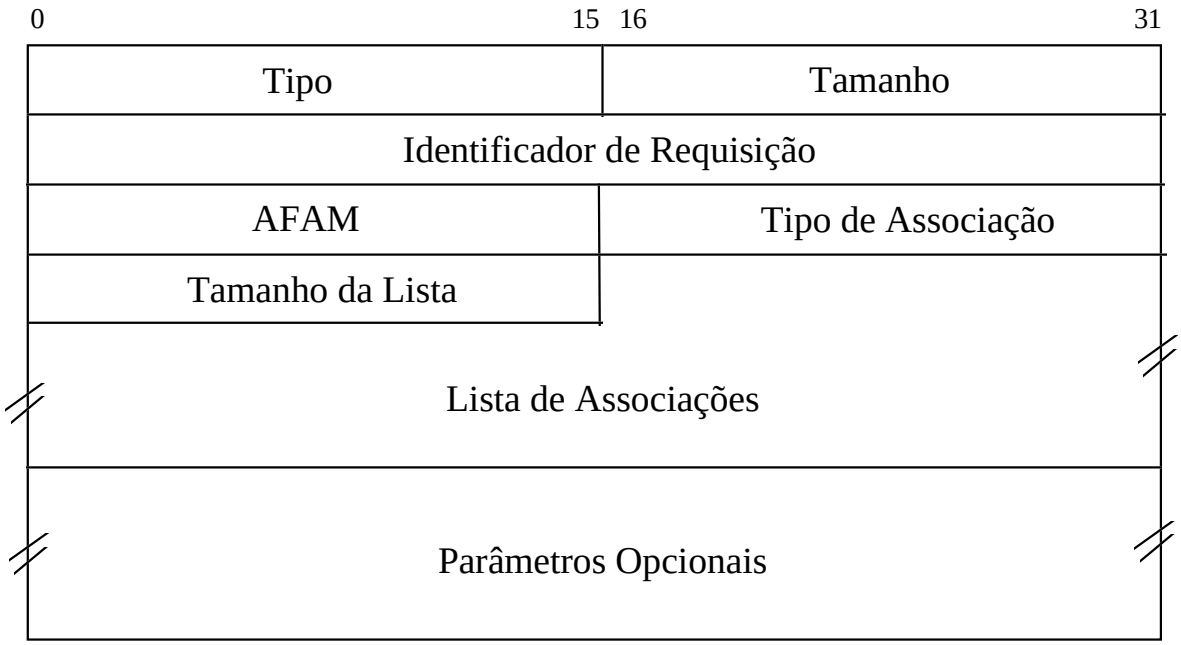


Figura 66: PIE tipo BIND utilizado para propagar associações.

O campo Tipo possui valor 200 (hexadecimal). O campo Tamanho carrega o tamanho total em bytes do PIE menos 4. Identificador de Requisição é utilizado para associar um BIND a um REQUEST_BIND. O campo AFAM (Address Family

Numbers) especifica a família de endereço de rede presente na associação (por exemplo, família Internet). O campo Tipo de Associação (BLIST_TYPE) contém um dos sete valores descritos acima. O campo Tamanho da Lista determina o tamanho da lista de associações que se segue. O campo Lista de Associações contem as associações que o TSR deseja propagar. Finalmente, o campo Parâmetros Opcionais contem parâmetros no formato tipo-tamanho-valor.

Para associações tipo 5 e 6 um elemento da lista de associações possui o formato dado pela Figura 67.

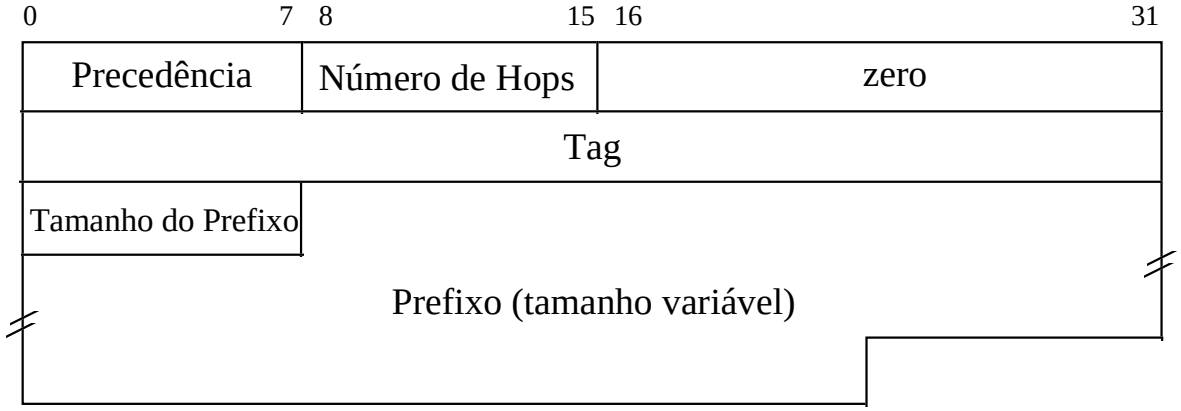


Figura 67: Associação tipos 5 e 6.

O campo Precedência determina a prioridade que o TSR irá atribuir ao tráfego definido por esta associação. Número de Hops informa o número de nós (*hops*) até o destino. O campo Tag contém o valor do tag para esta associação. O campo Tamanho do Prefixo contém o tamanho em bits do prefixo que representa a subrede de destino. Finalmente, o campo Prefixo contém o endereço da subrede de destino. Um enchimento de bits deve ser utilizado caso a mensagem TDP não seja múltiplo de 8 bits.

3.4.4 Utilização de Tag Switching com ATM

Um equipamento ATM operando como Tag Switching Router (ATM-TSR) não necessita de roteamento ou sinalização padronizados pelo ATM Forum. ATM-TSRs conectados utilizam VPI=0 e VCI=32 para propagar mensagens TDP encapsuladas no formato LLC/SNAP.

Tags são codificadas no campo VCI da célula ATM, podendo o campo VPI ser utilizado para transportar vários tags por célula. Durante o estabelecimento de sessão TDP, os ATM-TSR informam o intervalo de valores para VPI e VCI que são capazes de processar. A alocação de tags é sempre *downstream* por demanda.

Seja a topologia da Figura 60. Quando um roteador de borda recebe um pacote IP desprovido de tag, o mesmo seleciona o próximo *hop* da rota. Suponha que este *hop* está conectado via interface ATM e é um ATM-TSR. Caso a TIB do roteador não disponha de associação para o destino do datagrama, o roteador de borda envia uma mensagem TDP tipo REQUEST_BIND[15], recebendo como resposta uma associação tag-destino para esta rota. O campo Número de Hops da associação (Figura 67) contém o número de nós até o destino, sendo que o roteador pode utilizar este valor para decrementar o campo Tempo de Vida (TTL) do datagrama. O roteador então envia o datagrama ao ATM-TSR (encapsulado na AAL-5) no canal VCI=tag.

Quando, por ação de roteamento, o roteador de borda decide mudar uma dada rota (isto é, trocar o próximo *hop*) e possui associações com um ATM-TSR (o próximo *hop* que mudou), o roteador deve descartar as associações obtidas deste ATM-TSR e notificá-lo que tais associações não são mais necessárias[16].

Quando um ATM-TSR recebe uma requisição de associação para determinada rota o mesmo aloca um tag (VCI disponível), cria uma entrada em sua TIB para esta rota, mesmo que a rota já possua entrada na TIB. O ATM-TSR solicita uma associação para esta mesma rota ao seu próximo *hop*. Neste instante, o ATM-TSR pode agir de duas maneiras:

1. retornar imediatamente o tag alocado para o TSR que requisitou, colocando um valor reservado no campo Número de Hops da associação que significa "desconhecido";

2. esperar o recebimento da associação do ATM-TSR abaixo, para então retornar a associação com o campo Número de Hops incrementado de uma unidade em relação àquele recebido do ATM-TSR abaixo.

   Caso o ATM-TSR abaixo não seja capaz de criar uma associação, o ATM-TSR que recebeu a requisição deve destruir a associação criada localmente e informar o requisitante da impossibilidade de criar associação para esta rota.

A razão do ATM-TSR criar um novo tag para um mesmo destino, mesmo que tal destino já exista em sua TIB, se deve ao fato da camada de adaptação número 5 (AAL-5) não permitir o entrelaçamento de células durante a transmissão de um CPCS-PDU. Suponha que um ATM-TSR distribuisse o mesmo tag associado a determinado destino para vários TSRs acima. Caso dois ou mais TSRs, utilizando o mesmo VPI, transmitissem um pacote para o mesmo destino[17], teríamos celulas de mesmo par VPI/VCI mas pertancentes a CPCS-PDUs diferentes. A camada de convergência para a AAL-5 não permite tal situação.

[15] Supondo que uma sessão TDP já foi estabelecida.
[16] Mensagens TDP tipo WITHDRAW_BIND são empregadas para esta finalidade.
[17] Utilizando portanto o mesmo tag, isto é, o mesmo VCI.

3.4.5 Tag Switching: Vantagens e Desvantagens

Tag Switching tem como vantagem sua instalação imediata no vasto parque de roteadores Cisco em operação[18]. A definição flexível de fluxo é outra característica importante desta tecnologia. Outro ponto favorável é a diminuição do número de roteadores vizinhos em relação à utilização da nuvem ATM como infra-estrutura de camada 2 (agora uma chave ATM é um *peer* de rede). Entretanto, algumas desvantagens devem ser consideradas:

- ainda não existem implementações que possam validar a tecnologia;
- todos os nós operam como roteadores: cada nó necessita processar os protocolos de roteamento, diminuindo sua eficácia como elemento de comutação de pacotes;
- a utilização de vários VCI (tags) por rota é uma outra forma de aumentar o número de conexões (problema das N^2 conexões);
- a maneira de prover qualidade de serviço (QoS) ainda é incerta.

[18] A Cisco Systems é lider mundial neste segmento.

3.5 IP Navigator

IP Navigator é um produto da Cascade Communication Corp. na linha do modelo *peer*. Tal qual IP Switching e Tag Switching, IP Navigator integra funções de chaveamento (camada 2) e roteamento (camada 3) no mesmo equipamento de rede. O mercado alvo deste produto é WANs operando o protocolo IP, notadamente os grandes provedores de acesso Internet. Esta solução é proprietária pois o IP Navigator é uma extensão do produto Virtual Network Navigator (VNN) da Cascade, um software de gerência e operação das chaves Cascade de acesso (não necessariamente chaves ATM). Isto não quer dizer que o IP Navigator não seja capaz de interoperar com outras soluções, conforme mostraremos na sequência.

IP Navigator opera numa estrutura dada pela Figura 68.

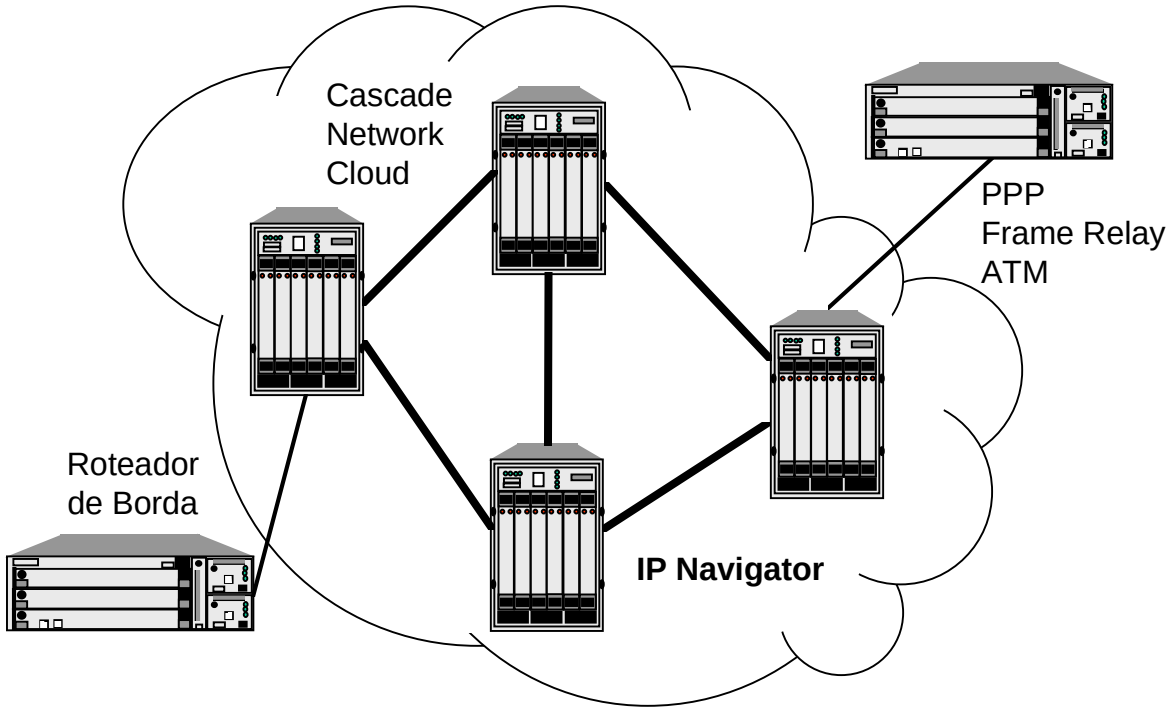


Figura 68: Topologia típica para o IP Navigator.

Algumas características importantes do IP Navigator são:

- elimina roteamento *hop-by-hop* no interior da nuvem (roteamento nível 3 se dá apenas nos roteadores de borda de entrada e de saída);
- resolve o problema das N^2 conexões que dificulta a escalabilidade das WANs;

- cada datagrama IP é comutado no interior da nuvem, independente de fluxo, tag, etc.;
- provê QoS, multicast e redes privadas virtuais (VPN: Virtual Private Network).

3.5.1 Conexões Multiponto-Ponto

Um conceito chave no IP Navigator é a conexão multiponto-ponto. Neste tipo de conexão cada chave no interior da nuvem se conecta às demais, como numa conexão ponto-multiponto. A diferença é que em conexões ponto-multiponto o tráfego se dá do nó raiz para os nós terminais (folhas). Em conexões multiponto-ponto o tráfego se dá dos nós terminais para o nó raiz. Cada chave mantém uma conexão multiponto-ponto com as demais, garantindo-se assim conectividade plena entre N chaves com N conexões. Este tipo de conexão não é suportado por nenhuma versão da UNI (inclusive a 4.0) sendo viável apenas em chaves Cascade através do VNN.

A Figura 69 ilustra uma conexão multiponto-ponto e como esta resolve o problema das N^2 conexões presente em arranjos *full mesh*.

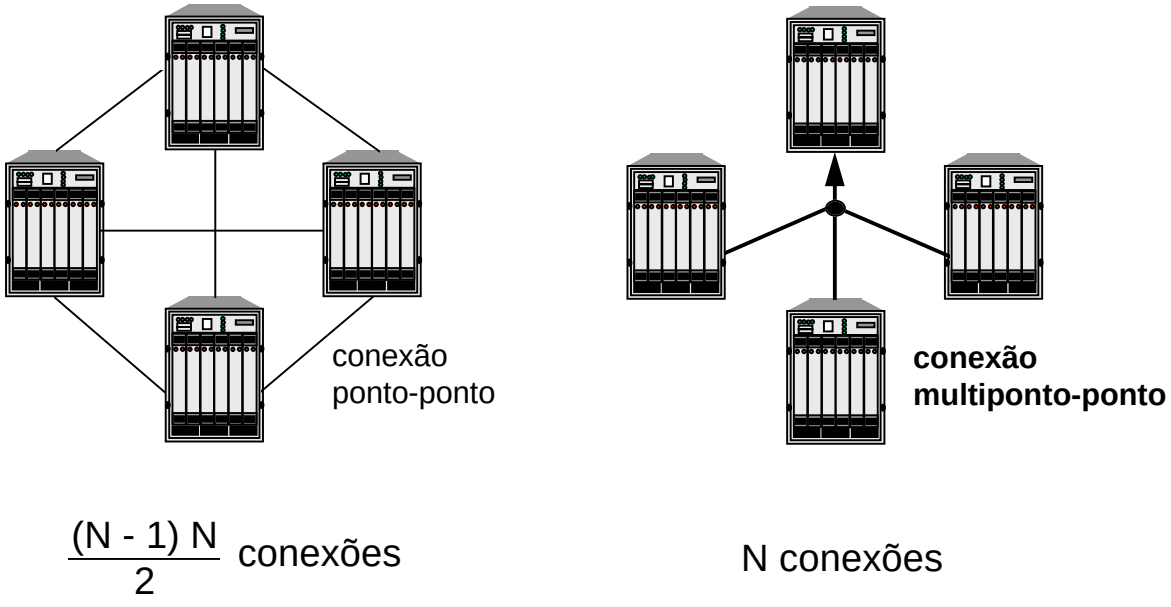


Figura 69: Conexão multiponto-ponto.

Conexões multiponto-ponto permitem que as ações de roteamento se concentrem apenas nas bordas da nuvem. O roteador de borda recebe um pacote IP e executa o roteamento de forma convencional. O próximo *hop* é sempre outro roteador de borda à distância de uma unidade lógica (um *hop*). O roteador então identifica a

conexão multiponto-ponto centrada no roteador de saída escolhido e envia o pacote por esta conexão. Recebido o pacote no roteador de saída o mesmo processa o roteamento também de forma convencional. A Figura 70 ilustra este processo. IP Navigator suporta os seguintes protocolos de roteamento:

- BGP-4 (Border Gateway Protocol v. 4);
- OSPF (Open Shortest Path First);
- RIP-2 (Routing Information Protocol v. 2);
- Roteamento estático.

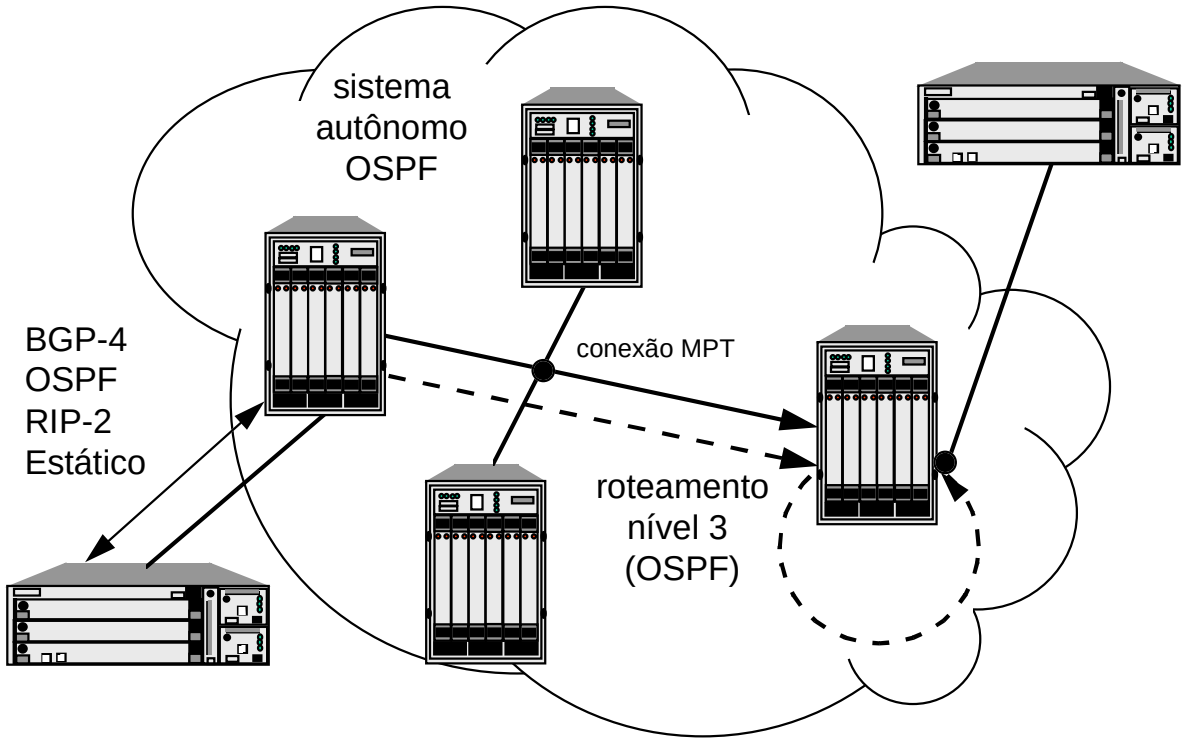


Figura 70: Roteamento no IP Navigator.

IP Navigator distribui as rotas no interior da nuvem de três maneiras:

1. distribuição plena (via OSPF): as rotas que uma chave "aprende" [19] são distribuidas para todas as chaves (ie, todas as chaves armazenam tabelas de roteamento completas - interior e exterior);

2. distribuição parcial (via IBGP): apenas chaves atuando como "refletores" armazenam tabelas completas de roteamento (as demais interagem com os refletores quando necessário);

[19] Via de regra como serultado da interação com os roteadores de borda a ela conectados.

3. distribuição de rotas criadas estaticamente (via OSPF): em geral estas rotas se destinam à *backbones* superiores ou redes privadas virtuais.

3.5.2 Qualidade de Serviço (QoS)

As conexões multiponto-ponto operam segundo o critério de "melhor esforço" (isto é, sem QoS). IP Navigator é capaz de prover QoS através da filtragem de determinado tipo de tráfego. A filtragem pode se dar por configuração ou por requisição através do protocolo RSVP. Os parâmetros do filtro são os mesmos do IP Switching: endereço IP de destino ou aplicação (endereço, protocolo, port). O tráfego filtrado utiliza um circuito Frame Relay entre chaves de ingresso e saída. Este circuito provê taxa garantida e atraso mínimo. Futuramente (atualmente ?) o tráfego filtrado será mapeado em conexões ATM tipo CBR, VBR-rt, etc., ou em conexões multiponto-ponto capazes de garantir QoS.

3.5.3 VPN (Virtual Private Network)

Redes privadas virtuais (VPNs) são subredes lógicas dentro de uma WAN formadas para atender necessidades específicas de clientes corporativos. VPNs são implementadas através de tabelas de roteamento customizadas para determinadas famílias de endereçs IP. Por exemplo (Figura 71) uma empresa deseja concentrar todo o tráfego Internet em sua sede para que o mesmo passe por um *firewall*. Neste caso, o IP Navigator permite rotear todo o tráfego IP para a sede via roteamento forçado (isto é, roteamento definido por ação de gerência, não por ação dos protocolos de roteamento).

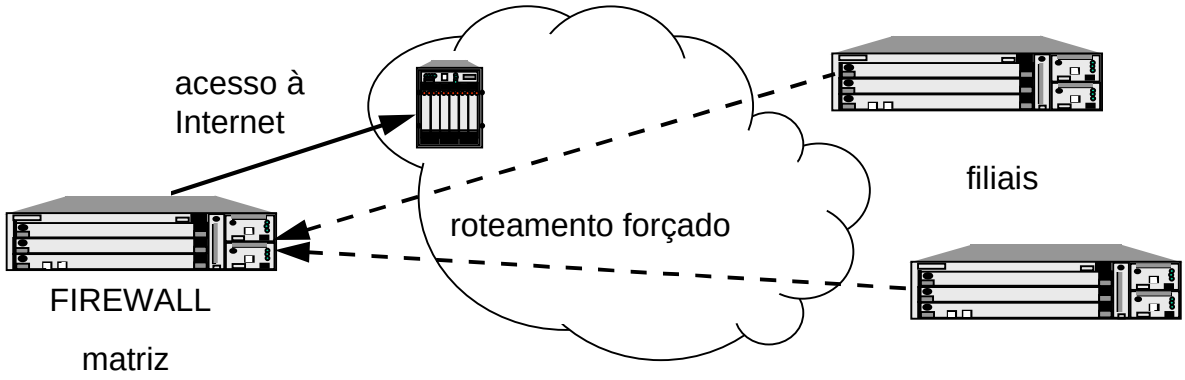


Figura 71: Implementação de rede privada virtual (VPN) via roteamento forçado.

3.5.4 Interoperabilidade

IP Navigator, apesar de solução fechada, possui "ganchos" de interoperabilidade. À estes "ganchos" é permitido conectar via circuito Frame Relay, ATM ou PPP (Figura 72):

- um Provedor de Acesso Internet via protocolo de roteamento BGP-4;
- um cliente corporativo via protocolo de roteamento OSPF (utilizando ou não VPN);
- uma infra-estrutura baseada em IP Switching via protocolo IFMP (Ipsilon Flow Management Protocol);
- uma infra-estrutura baseada em Tag Switching via protocolo TDP (Tag Distribution Protocol);
- uma nuvem ATM executando MPOA via protocolo NHRP.

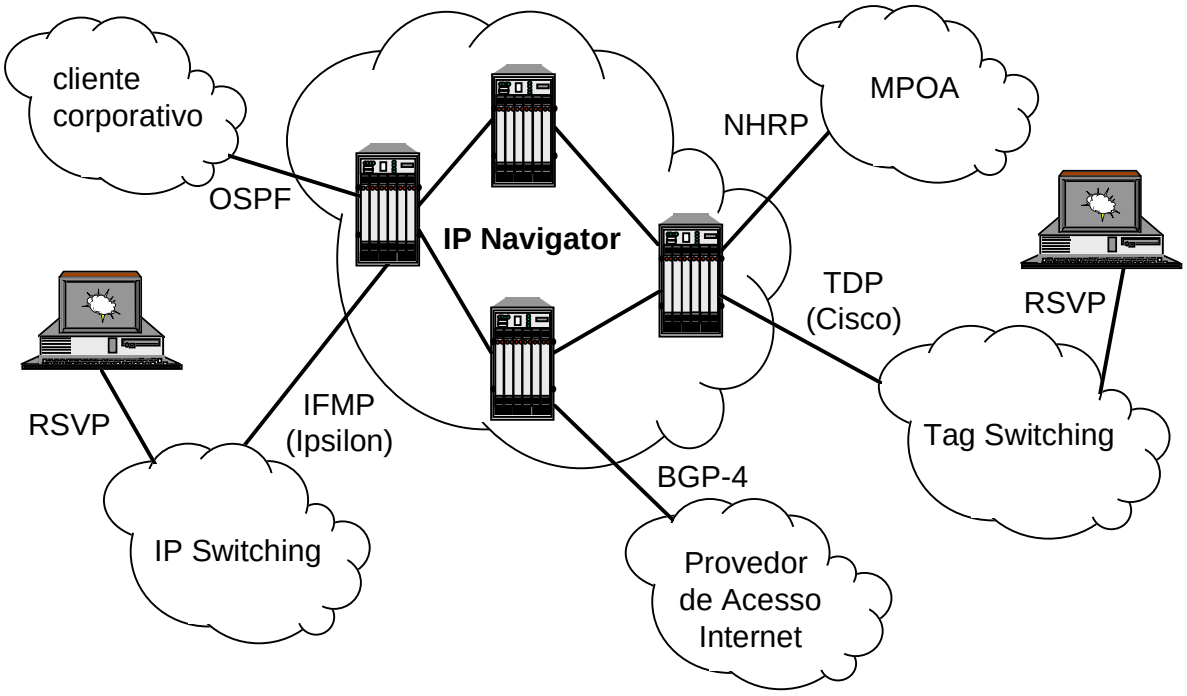


Figura 72: Interoperabilidade do IP Navigator.

3.5.5 IP Navigator: Vantagens e Desvantagens

IP Navigator possui a vantagem de ser um produto disponível comercialmente. Como tecnologia, equaciona adequadamente o problema das N^2 conexões, além de não necessitar de uma definição de fluxo, exceto, eventualmente, para fins de provimento de qualidade de serviço. IP Navigator também interopera com outras

tecnologias, inclusive IP Switching e Tag Switching. Redes privadas virtuais é outra característica interessante do IP Navigator.

Como desvantagens, temos:

- é uma solução proprietária pois necessita do VNN (Virtual Network Navigator) presente apenas em produtos Cascade;
- utiliza Frame Relay para provimento de qualidade de serviço;
- todas as conexões multiponto-ponto devem ser atualizadas sempre que uma chave entra ou sai de operação (pode ser ineficiente para grandes WANs);
- disponível apenas em chaves ATM de acesso de alta capacidade da Cascade sendo inviável economicamente para redes corporativas.

4 Novos Protocolos INTERNET[20]

4.1 IP Versão 6 (IPv6) Sobre ATM

4.1.1 Introdução

Esta seção pretende apresentar um *overview* das principais características da versão 6 do protocolo IP (IPv6, também conhecido por IPng - IP next generation) e de sua integração com a tecnologia ATM (Asynchronous Transfer Mode). O IPv6 vem como uma proposta de evolução sobre a atual versão do IP (IPv4) em operação no mundo todo. A nova versão procura manter a arquitetura e filosofia da versão atual, atacando principalmente os pontos críticos de endereçamento e roteamento e tratando de assuntos emergentes como segurança, autoconfiguração, serviços de tempo-real (multimedia) e a transição de uma versão para outra. Outro aspecto importante é a operação do IPv6 sobre as redes ATM.
A primeira proposta do IPv6 foi apresentada pelo IETF em julho de 94 na RFC 1752 e aprovada pelo IESG em novembro do mesmo ano. Em dezembro de 95 o protocolo IPv6 foi apresentado à comunidade Internet através de RFCs, tendo como principais autores Steve Deering e Robert Hinden. O crescimento da Internet foi o ponto principal que motivou a necessidade de uma revisão no protocolo IP. A taxa de crescimento da Internet tem sido exponencial, dobrando a cada 12 meses.

A ameaça da escassez de endereços baseia-se em 2 fontes distintas de informação. A RFC-1715 propõe o chamado fator H para medida da eficiência de atribuição de endereços:

$$H = \frac{\log(\text{no. de endereços})}{\text{no. de bits}}$$

Na prática, detectou-se que a taxa de atribuição efetiva gira em torno de H=0,22 a H=0,26, ou seja, 32 bits podem endereçar de 11 a 200 milhões de máquinas diferentes. Hoje, 32 bits seriam suficientes para endereçar todos os computadores existentes, mas deixaria uma margem de folga pequena que prevê-se esgotar entre 2005 e 2015 se as taxas atuais de crescimento se mantiverem.

Por outro lado, a tendência é de que outros setores, além dos tradicionais, comecem a se conectar à Internet. O mercado de computação móvel tende a crescer muito, ameaçando substituir os atuais celulares e pagers. O surgimento da TV interativa, do vídeo sob demanda e das HDTVs apontam para a realidade de termos aparelhos de TV como hosts na Internet. Outro mercado que poderia usufruir desses recursos é o de controle de aparelhos do dia-a-dia, como iluminação, motores, aquecimento e ar condicionado, entre outros.

[20] Baseado na referência [Wad96].

Esses mercados potenciais apontam para um aumento da taxa de crescimento da Internet e demandam funcionalidades atualmente não suportadas pelo IP, como autoconfiguração, baixo *overhead*, autenticação e confidencialidade. O IPv6, ao contemplar as necessidades destes novos mercados, apresenta-se como plataforma unificadora de uma rede de nível mundial, interoperável, baseada em protocolos não proprietários.

Em linhas gerais o protocolo IPv6 possui as seguintes características:

- tamanho do endereço aumentado de 32 (Ipv4) para 128 bits (IPv6);
- cabeçalho:

⇒ 20 bytes no IPv4, 40 no IPv6;
⇒ 12 campos obrigatórios no IPv4, 8 no IPv6;
⇒ Ausência de checksum no IPv6;
⇒ Extensões no cabeçalho (opções no IPv4).

- suporte ao conceito de fluxo;
- tratamento mais eficaz à segurança e autenticação;
- fragmentação de datagramas apenas na origem.

4.1.2 IPv6: Terminologia

A Figura 73 ilustra os principais conceitos associados com o IPv6.

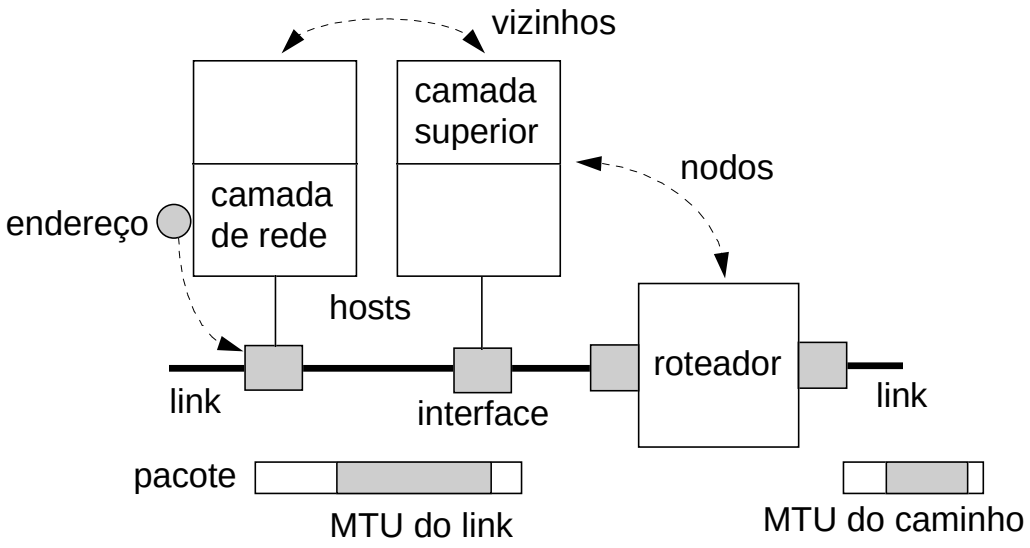


Figura 73: Termos associados ao IPv6.

Nodo: um dispositivo que implementa o IPv6;

Roteador: um nodo que propaga pacotes IPv6 que não sejam explicitamente endereçados ao nodo;
Host: qualquer nodo que não roteador;
Camada Superior: uma camada que implementa um protocolo imediatamente acima do IPv6 (exemplo:
TCP, UDP, ICMP);
Link: uma infra-estrutura que permite dois nodos se comunicar utilizando um protocolo imediatamente abaixo do IPv6 (por exemplo: Ethernet);
Vizinhos: nodos conectados a um mesmo *link*;
Endereço: um identificador utilizado pelo IPv6 para identificar uma interface ou um grupo de interfaces;
Pacote: um cabeçalho IPv6 mais carga;
MTU (Maximum Transfer Unit) **do Link**: tamanho máximo de um pacote em bytes que pode ser transportado numa única ação de comunicação sobre um *link* (exemplo: 1500 bytes para Ethernet);
MTU do Caminho: o menor MTU de todos os *links* que compõem o caminho.

4.1.3 Estrutura do Protocolo IPv6

A Figura 74 ilustra o cabeçalho mínimo de um pacote IPv6.

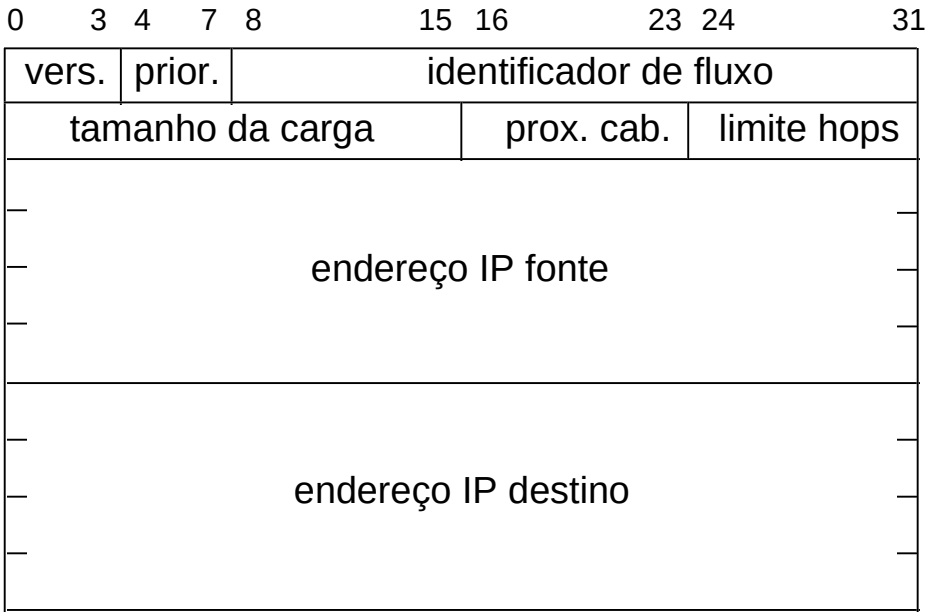


Figura 74: Pacote IPv6.

A idéia inicial era utilizar o campo de Versão do cabeçalho para diferenciar pacotes entre IPv4 e IPv6. Essa idéia foi abandonada ao se constatar que diversos

fabricantes não se valiam dessa informação, simplesmente ignorando-a. Ficou resolvido, então, que sempre que possível os pacotes IPv6 serão demultiplexados na camada 2. No caso, o Ethertype associado ao IPv6 é **86DD** (hexadecimal), enquanto que o Ethertype do IPv4 é 8000 (hexadecimal).

Quando se comparam os cabeçalhos das duas versões do IP, nota-se que o novo cabeçalho é bem mais simples que seu antecessor, apesar de possuir 40 bytes contra 20 na versão anterior. Vários campos foram removidos ou passaram a ser extensões. Dentre as simplificações sobre o modelo anterior, temos:

Formato fixo para o cabeçalho

A remoção de campos opcionais de tamanho variável do cabeçalho agiliza seu processamento, aumentando a performance dos dispositivos de roteamento, e obsoleta a necessidade do campo de comprimento do cabeçalho.

Remoção do *checksum* do cabeçalho

Embora pareça inseguro, as conseqüências de se remover o *checksum* do cabeçalho do novo IPv6 são mínimas, já que a maioria dos procedimentos de encapsulação usam *checksum* (incluindo a camada de adaptação do ATM).

Remoção de identificação de segmentação

No IPv6, os hosts primeiro têm de descobrir o MTU do caminho antes de enviar um pacote (o tamanho default mínimo de 576 bytes continua válido). Como pacotes maiores que este valor são descartados pela rede, não existe a necessidade de se fragmentar o pacote na rota. Existe, entretanto, a opção de fragmentação fim-a-fim, via cabeçalhos de extensão.

Remoção de campos opcionais dentro do cabeçalho

O IPv6 separou os campos de opção em cabeçalhos próprios chamados de cabeçalhos de extensão. Estas mudanças feitas na forma em que os campos opcionais são inseridos no pacote tornou seu uso mais eficiente, menos restritivo, permitindo maior flexibilidade e expansões futuras.

Extensão do espaço de endereçamento

O IPv6 expandiu o tamanho dos endereços de 32 para 128 bits, suportando assim mais níveis de hierarquia de endereços, maior número de hosts endereçáveis e possibilidade de autoconfiguração simplificada. A escalabilidade de multicast foi melhorada através da adição de um campo de escopo. Foi criado um novo tipo de endereçamento chamado de Anycast para identificar um conjunto de hosts [(2)] onde

[(2)] A rigor, um conjunto de interfaces.

o pacote pode ser entregue a qualquer host deste conjunto, permitindo controle de rotas do tipo *source route* (roteamento na origem).

Suporte ao conceito de fluxo
Foi adicionada uma nova funcionalidade que permite identificar fluxos específicos de tráfego que necessitem de qualidade de serviço, como por exemplo aplicações de tempo real.

Autenticação e privacidade

O IPv6 prevê a definição de extensões para fins de autenticação, integridade de dados e confidencialidade.

O campos do cabeçalho de um pacote Ipv6 possuem o seguinte significado

Versão: versão corrente (6);
Prioridade: prioridade deste datagrama em relação aos demais emitidos pela mesma fonte:

- 0 - 7: tráfego que provê controle de congestionamento (exemplo: TCP/IP);
- 8 - 15: tráfego de "tempo real" (exemplo: RTP).

Identificador de fluxo: utilizado para identificar um "fluxo": seqüência de datagramas emitidos por uma mesma fonte que transportam dados de tempo-real (exemplo: amostras de áudio ou vídeo). Estes datagramas recebem tratamento especial por parte dos roteadores.
Tamanho da carga: tamanho do datagrama em bytes, excluindo-se o cabeçalho padrão.
Próximo Cabeçalho: identificado do cabeçalho de extensão que se segue.
Número de Hops: número máximo de roteadores que o datagrama pode percorrer (equivalente ao campo "Tempo de Vida" do IPv4);
Endereço IP fonte e destino: endereços de origem e destino do datagrama.

4.1.4 Cabeçalhos de Extensão

Os campos opcionais do IPv6 são apresentados em cabeçalhos de extensão, colocados entre o cabeçalho principal e os cabeçalhos dos pacotes da camada superior, formando um verdadeiro aninhamento de cabeçalhos (Figura 75). A maioria dos cabeçalhos de extensão do IPv6 não são examinados pelos roteadores durante o tráfego dos pacotes até atingir o destino final, aumentando a performance do roteamento no caso da existência de campos opcionais.

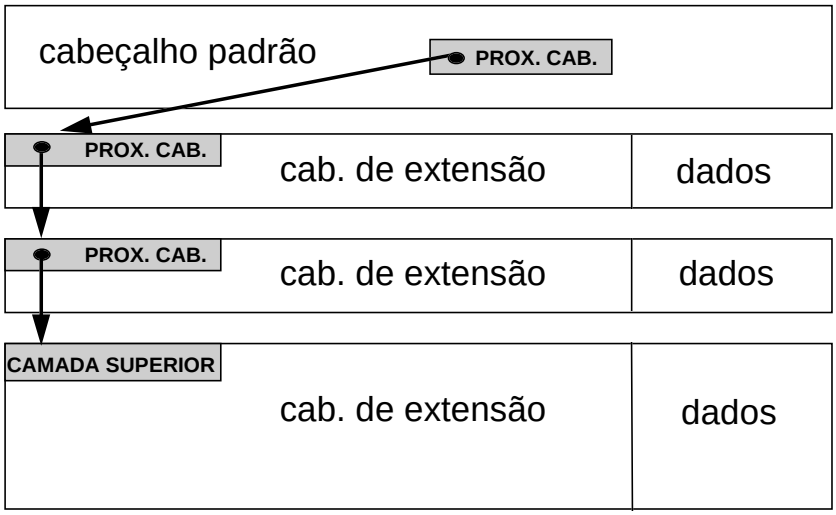


Figura 75: Estrutura de cabeçalhos do IPv6.

Ao contrário dos campos opcionais do IPv4, cabeçalhos de extensão podem ser de qualquer tamanho e o total de campos opcionais não está mais restrito a 40 bytes. Desta forma, abre-se a possibilidade de uso de características inexploráveis no IPv4, como as opções de autenticação e encapsulação de segurança.

De forma a agilizar o processamento dos cabeçalhos de extensão e do pacote de transporte que segue, os campos opcionais do IPv6 são sempre múltiplos de 8 octetos.

Cabeçalhos de extensão são geralmente tratados no destino (exceção: hop-by-hop). A RFC 1700 determina a identificação de protocolos presentes nos cabeçalhos de extensão (campo prox.cab.). Exemplo:

0 - cabeçalho hop-by-hop
1 - cabeçalho do protocolo ICMP
6 - cabeçalho do protocolo TCP
44 - cabeçalho de fragmentação
88 - cabeçalho do protocolo OSPF
59 - NULL (último cabeçalho)

A Figura 76 ilustra um pacote IPv6 dotado de cabeçalho de extensão.

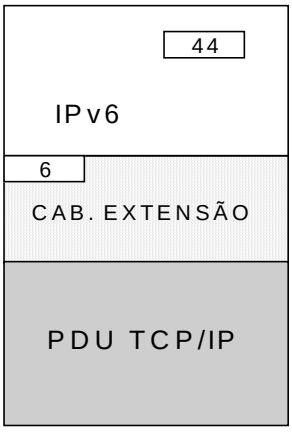


Figura 76: Pacote IPv6 com cabeçalho de extensão.

A Tabela 3 ilustra os tipos de cabaçalhos de definidos.

0	hop-by-hop	tratamento em todos os roteadores e no destino
60	destino (*)	tratamento apenas no destino
43	roteamento	estabelecimento de rota na origem
44	fragmentação	fragmentação na origem
51	autenticação	autenticação do emissor, integridade do datagrama
52	segurança	criptografia do datagrama
...	camadas superiores (TCP, ...)	

(*): pode ocorrer mais de uma vez no datagrama

Tabela 3: Tipos de cabeçalho de extensão.

4.1.5 Endereçamento

Endereços IPv6 são identificadores de interfaces ou de conjuntos de interfaces formados por 128 bits, agrupados em 3 tipos distintos :

ENDEREÇOS UNICAST: identificam uma única interface. Um pacote enviado a um endereço unicast é entregue à única interface identificada por este endereço.

ENDEREÇOS ANYCAST: identificam um conjunto de interfaces. Um pacote enviado a um endereço anycast é entregue a qualquer uma das interfaces identificadas por este endereço, tipicamente a mais "próxima" segundo o algoritmo de roteamento.

ENDEREÇOS MULTICAST: identificam um conjunto de interfaces. Um pacote enviado a um endereço multicast é entregue a todas as interfaces identificadas por este endereço.

O IPv6 suporta diversos desses tipos de endereço na mesma interface. Isto significa que uma interface pode pertencer a múltiplos domínios, por exemplo, ter ligações com diferentes provedores de acesso à Internet. Endereços IPv6 são representados por oito grupos de quatro números hexadecimais, conforme ilustra a Figura 77.

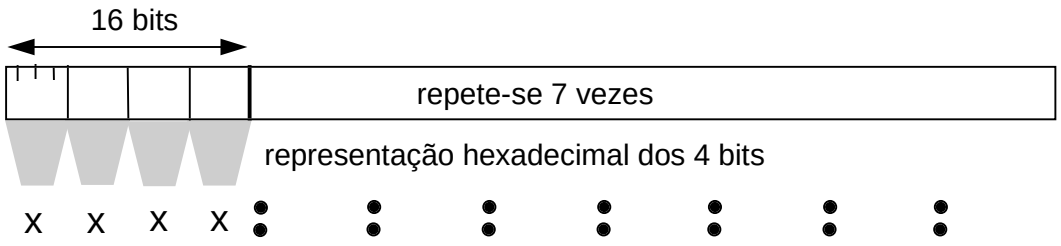


Figura 77: Estrutura de um endereço IPv6.

Os projetistas do IPv6 resolveram representar os endereços IPv6 segundo uma nova notação de 8 inteiros de 16 bits separados por dois pontos:

FEDC:BA98:7654:3210:ABD4:3280:09AF:0001

São usados números hexadecimais devido a seu caráter compacto e direto, e abreviações foram introduzidas de forma a facilitar seu uso por parte dos administradores de rede. Primeiro, grupos de zeros podem ser agrupados ou suprimidos (quando à esquerda). O número IPv6

FEDC:0000:0002:0000:0000:0000:0000:B2EF

pode ser escrito como:

FEDC:0:2:0:0:0:0:B2EF

Além disso, é introduzida a notação dos dois-pontos duplos, que casa com qualquer seqüência de 0s consecutivos (uma única vez em cada notação de forma a evitar ambigüidades). O endereço acima poderia ser escrito como:

FEDC:0:2::B2EF

Alguns endereços são reservados:

0:0:0:0:0:0:0:0 (**::**) endereço não especificado;
0:0:0:0:0:0:0:1 (**::1**) loopback.

IPv6 define alguns tipos de endereço dado pelo prefixo (bits de mais elevada ordem) conforme ilustra a Tabela 4:

tipo	prefixo
NSAP	0000 001
IPX	0000 010
centrado no provedor	010
geográfico	100
local ao link	1111 1110 10
local ao site	1111 1110 11
multicast	1111 1111

Tabela 4: Tipos de endereços IPv6.

Endereços Unicast

O IPv6 introduz diversas formas de se identificar univocamente uma interface. Endereço unicast de um provedor global, endereço unicast geográfico, endereço NSAP, endereço IPX hierárquico, endereço site-local, endereço enlace-local e os endereços do IPv4.

Os endereços unicast de provedor global são usados para comunicação global, da mesma forma como os atuais endereços IPv4 atribuídos pelo CIDR. A Figura 78 ilustra o formato deste endereço.

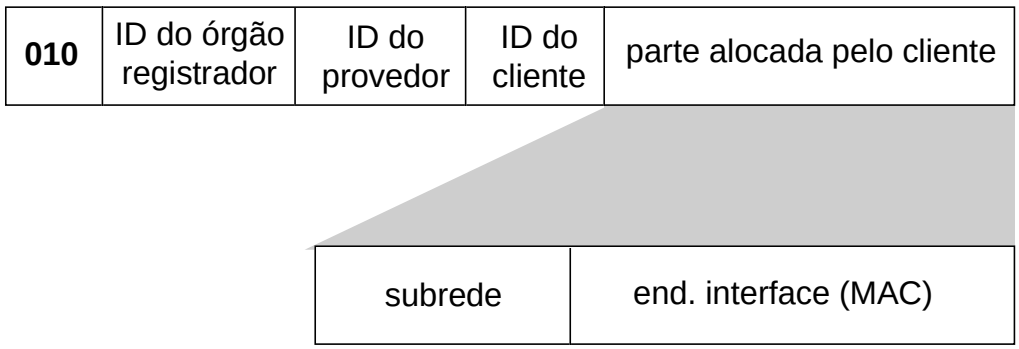


Figura 78: Endereço unicast centrado no provedor de acesso.

Existem 2 tipos de endereços de uso local, os locais ao link e os locais ao *site* (**Error! Reference source not found.**). Os endereços locais ao *link* podem ter unicidade local ou global, mas são roteados apenas dentro da subrede, só podendo ser usados dentro do mesmo *link*. Os endereços locais ao *site* tem escopo restrito ao *site* (rede privativa).

local ao link:

1111111010	**0**	end. interface (MAC)

local ao site:

1111111011	**0**	subrede	end. interface (MAC)

Figura 79: Endereços unicast local ao link e local ao site.

Os endereços locais ao *link* foram planejados para serem usados para endereçamento dentro de um mesmo enlace de forma a suportar a autoconfiguração, o protocolo de Neighbor Discovery e roteamento quando não existirem roteadores.

Ambos os tipos de endereços são constituídos de um identificador (endereço) de interface que deve ser único no domínio. A escolha de um identificador de interface é um compromisso delicado. No mesmo enlace é preciso identificar, univocamente, todas as interfaces usando um espaço numérico que minimize a probabilidade de interfaces diferentes virem a ter o mesmo identificador. Por outro lado, o comprimento deste identificador deve ser o menor possível de forma a não comprometer a quantidade de bits disponíveis para formar a hierarquia de prefixos para roteamento. Um comprimento de 48 bits (6 octetos) foi adotado pelo padrão IEEE 802 e é o mais comum atualmente.

No ambiente ATM, podem existir interfaces IP lógicas sobre interfaces ATM lógicas, todas sobre uma mesma interface ATM física. Se estas interfaces IP lógicas pertencerem ao mesmo enlace IPv6, então cada uma precisará de um identificador de interface único de forma a gerar endereços local ao link diferentes.

A combinação de endereço de interface e subrede forma o endereço local ao *site* que tem dimensão suficiente para numerar todos os hosts de uma rede privada. Esse esquema permite, ainda, um avanço significativo em relação ao IPv4 no sentido de que ao se conectar uma rede privada à Internet, não existe a necessidade de alteração manual da numeração das máquinas - os mesmos números podem ser

usados em combinação com prefixos adequados do tipo ID do registrador, ID do provedor e ID do cliente.

O IPv6 também suporta a utilização dos endereços do tipo IPv4 em uso atualmente, de forma a facilitar o período de transição do IPv4 para o IPv6. Os atuais endereços de 32 bits são embutidos na estrutura de 128 bits do IPv6 de 2 formas diferentes: uma para hosts que implementam o IPv6 mas optam por usar o endereçamento do IPv4 (endereço IPv6 compatível com IPv4) e outra para hosts não preparados para suportar o IPv6 (endereço IPv6 mapeável em IPv4).

96 bits	32 bits
0 : 0 : 0 : 0 : 0 : 0	Endereço IPv4

Formato do endereço IPv6 compatível com IPv4

80 bits	16 bits	32 bits
0 : 0 : 0 : 0 : 0	FFFF	Endereço IPv4

Formato do endereço IPv6 mapeável em IPv4

Figura 80: Utilização do endereço IPv4 no IPv6.

Endereços Anycast

Os endereços anycast representam um conjunto de interfaces no qual o pacote é entregue para a interface mais próxima segundo a medida de distância do algoritmo de roteamento.

Os endereços anycast possibilitam as políticas de roteamento na fonte, isto é, um host pode escolher por onde seu tráfego deve fluir. Endereços anycast podem ser inseridos no cabeçalho de extensão de roteamento, fazendo com que o pacote seja entregue através de um provedor em particular ou de uma série de provedores.

Os endereços de anycast são alocados dentro do espaço de endereçamento unicast, usando um dos formatos definidos de endereçamento unicast (Figura 81). Endereços anycast são sintaticamente indistintos de endereços unicast, portanto quando um endereço anycast é atribuído aos hosts estes devem ser explicitamente informados de que se trata de um endereço de anycast.

subrede	**0**

Figura 81: Endereço IPv6 anycast.

Endereços Multicast

No IPv6 não existe mais o endereço de broadcast (.255). O conceito foi generalizado para endereços de multicast que identificam um conjunto qualquer de interfaces. Os endereços de multicast possuem um formato diferenciado e um pacote enviado a um endereço multicast é entregue a todas as interfaces identificadas por ele. Não existe restrição quanto ao número de endereços multicast associados a cada interface. A Figura 82 ilustra o formato de um endereço multicast.

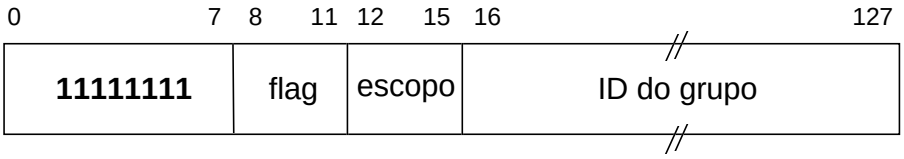


Figura 82: Endereço Ipv6 multicast.

O campo *flag* indica se o endereço é alocado permanentemente (0000) ou de forma transiente (0001). O campo *escopo* dita a cobertura do multicast:

1 - multicast local ao nodo
2 - multicast local ao link
5 - multicast local ao site
8 - multicast local à organização
E - multicast global
F - reservado

4.1.6 Autoconfiguração

Autoconfiguração significa que um host pode automaticamente descobrir e registrar os parâmetros que necessita para se conectar à Internet ou a qualquer outra rede. A autoconfiguração de endereços é parte integrante de todas as implementações de IPv6.

No caso da interface possuir endereço MAC, a autoconfiguração deve usar este endereço na composição de seu endereço de enlace-local. Como é possível atribuir vários endereços lógicos a uma mesma interface física, o endereço MAC só pode ser usado na primeira interface lógica e as demais seguirão um procedimento para endereços duplicados.

Se um endereço E.164 estiver disponível em vez do endereço MAC, então o endereço de enlace-local deverá ser composto dos 12 dígitos menos significativos do endereço e.164 codificados em BCD de forma a compor um endereço de 6

octetos. Se um endereço X.121 estiver disponível, então o endereço local ao *link* também será composto dos 12 dígitos menos significativos do endereço X.121 codificados em BCD de forma a compor os 6 octetos. Se um endereço ATMForum NSAP estiver disponível, o endereço local ao link deverá ser formado usando o campo ESI do NSAP.

Nos demais casos, incluindo quando for gerado um número repetido (duplicado), pode ser feita configuração manual, aleatória ou pode-se usar o DHCPv6.

A detecção de endereço duplicado é realizada com uma pequena melhoria no protocolo NHRP através de um bit de requisição de unicidade. Quando o NHS/NHC receber um pacote com este bit ativo, verifica em sua memória *cache* por conflitos, rejeitando o endereço caso ele já exista, ou registrando-o caso ele seja inédito.

A autoconfiguração é chamada de sem-estado (*stateless*) se o host sozinho puder definir os parâmetros para configurar sua interface. O host não precisa ter autonomia sobre todos os parâmetros, podendo existir o caso onde ao se juntar aos grupos de multicast, o host seja instruído a consultar um servidor de configuração de endereços que lhe fornecerá dados para a configuração da interface.

Após configurado o endereço de enlace-local da interface, o host se junta ao grupo de multicast pré-definido all-hosts (FF02::1) e em seguida envia uma mensagem de requisição ao grupo all-routers (FF02::2), iniciando o protocolo Neighbor Discovery ("descoberta dos vizinhos"), conforme ilustra a Figura 83. Roteadores respondem à solicitação (*Router Advertisement*) com o endereço de sua interface conectada ao *link*. O protocolo Neighbor Discovery (ND) usa o ICMPv6 para propagar suas mensagens.

Descobertos os roteadores, hosts iniciam um procedimento de descoberta de vizinhos. Mensagens ND tipo *Neighbor Solicitation* são propagadas no grupo FE02::01 e FE02:02. Hosts já configurados respondem com mensagem tipo *Neighbor Advertisement*.

O protocolo Neighbor Discovery é o procedimento do IPv6 que substitui tanto protocolos de resolução de endereço (ARP) quanto de descoberta de roteadores do IPv4.

Dois conceitos são introduzidos no Neighbor Discovery (Figura 84):

ON-LINK. Endereço associado à interface de um vizinho num enlace compartilhado. Um host considera que um endereço está *on-link* quando estiver coberto por um dos prefixos do enlace, ou quando um roteador especificar este endereço como destino numa mensagem de redireção de rota (*Redirect*) ou quando for recebida uma mensagem de anúncio de vizinho (*Neighbor Advertisement*) ou ainda quando for recebida uma mensagem de anúncio de roteador (*Router Advertisement*).

OFF-LINK. Oposto de *on-link*. Endereço não-associado a nenhuma interface conectada a um enlace compartilhado. Endereços *off-link* são considerados acessíveis apenas através de um roteador diretamente conectado ao enlace.

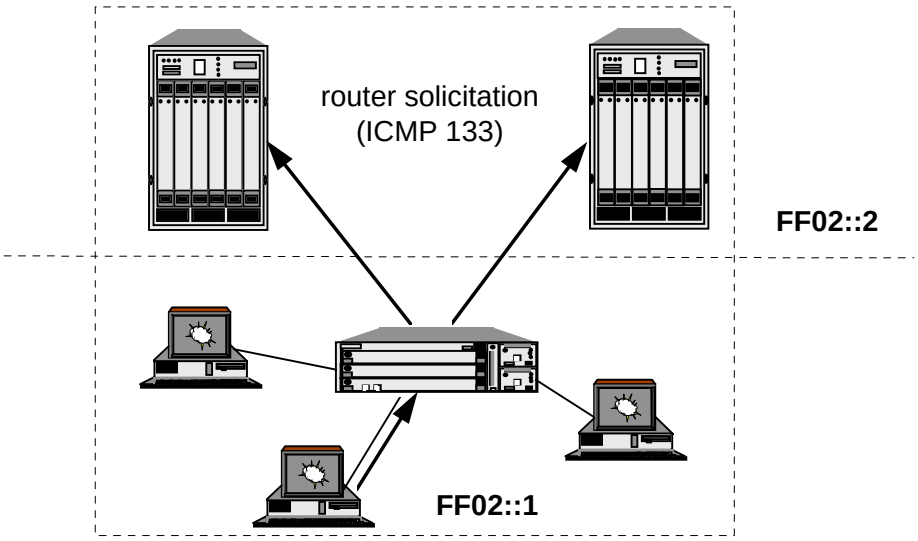


Figura 83: Protocolo Neighbor Discovery (ND) - identificação dos roteadores presentes na subrede.

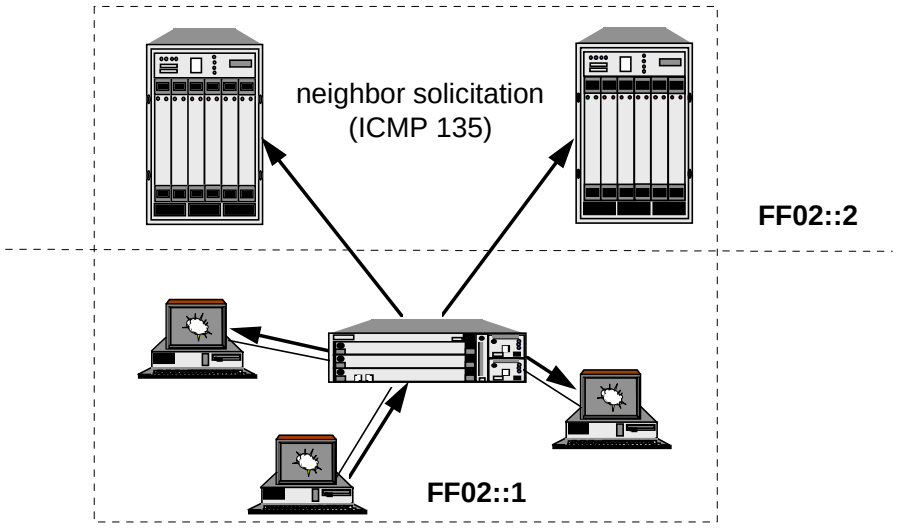


Figura 84: Protocolo Neighbor Discovery (ND) - identificação de hosts vizinhos.

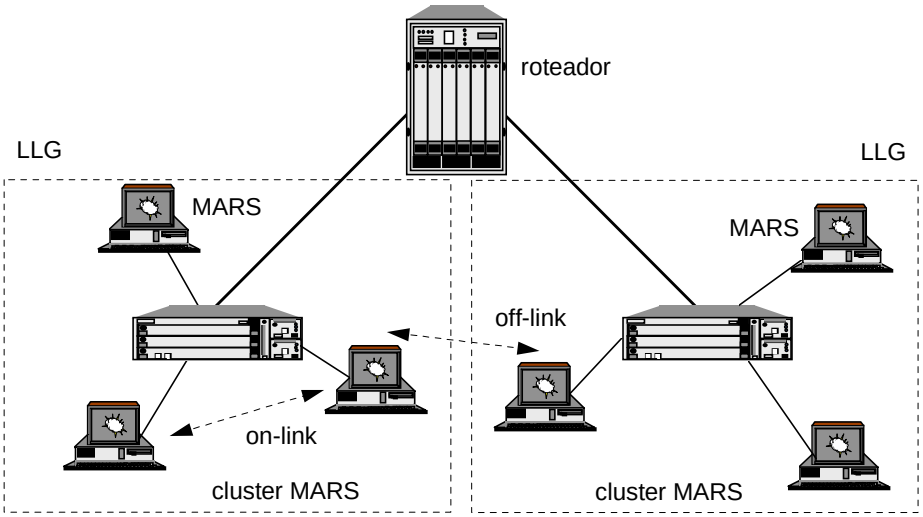


Figura 85: Protocolo ND: hosts *on-link* e *off-link*.

O ambiente ATM complica o sentido da palavra enlace da mesma forma com que complica o sentido de subrede. No caso do IPv4, é criado o conceito de LIS (Logical IP Subnet) como um conjunto administrativo de hosts que compartilham do mesmo prefixo de roteamento (máscara de rede e subrede).

[Arm96a] propõe, então, a criação do conceito de LLG (Logical Link Group - grupo de enlace lógico) para o caso do IPv6. O LLG consiste de hosts administrativamente configurados para estarem “em enlace” uns com os outros.

O Neighbor Discovery assume que suporte a multicast é trivialmente disponível na camada de enlace, o que não é verdade no caso do ATM. A emulação de multicast deve ser obtida com o protocolo MARS. conjuntos de hosts membros do mesmo cluster MARS seriam membros do mesmo LLG (Figura 85). Além disso, não faz referência clara ao conceito de conexão direta entre hosts (sem intermediação de um roteador) presente em protocolos como o NHRP.

A distinção para vizinhos *on-link* ou *off-link* feita pelo Neighbor Discovery leva a crer que conexões diretas só poderiam ser estabelecidas entre hosts da mesma LLG (como no IP clássico sobre ATM), mas para maximizar a eficiência da camada de enlace, tais conexões diretas deveriam ser suportadas no IPv6 também.

No caso do Neighbor Discovery sobre ATM, deve-se considerar o domínio LLG. O LLG é o mesmo conjunto de hosts que compõem o cluster MARS. Este é o conjunto inicial de vizinhos da interface e o endereço local ao link só precisa ser único dentro deste conjunto.

O recebimento de mensagens de anúncio de vizinhos ou outras operações semelhantes podem resultar na expansão do conjunto de vizinhos de uma interface.

Entretanto, isto não altera o conjunto de interfaces que forma seu LLG, levando a 3 tipos de relacionamento entre interfaces IPv6:

1. mesmo LLG, vizinhos;
2. fora do LLG, vizinhos (esta situação ocorre quando um host recebe uma mensagem de redireção por parte do roteador indicando a interface ATM do destino em outro LLG);
3. fora do LLG, não-vizinhos.

A comunicação entre hosts vizinhos num mesmo LLG se dá por estabelecimento de conexão ATM ponto-ponto e encapsulamento LLC/SNAP do pacote IPv6 nesta conexão (Figura 86).

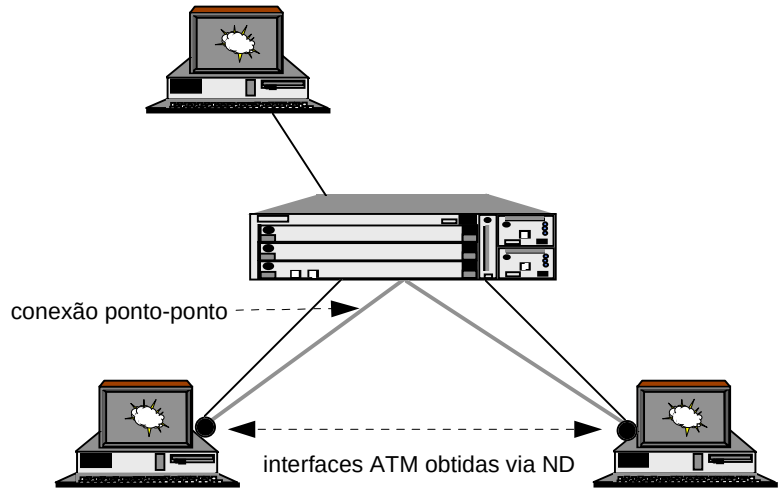


Figura 86: Comunicação entre hosts vizinhos num mesmo LLG.

A comunicacão entre hosts vizinhos em LLGs distintos também se dá por conexão ATM ponto-multiponto estabelecida após uma mensagem ND de redirecionamento de fluxo indicando a interface ATM do destino em outro LLG. Neste caso os hosts vizinhos são ditos *transientes* (figura 87).

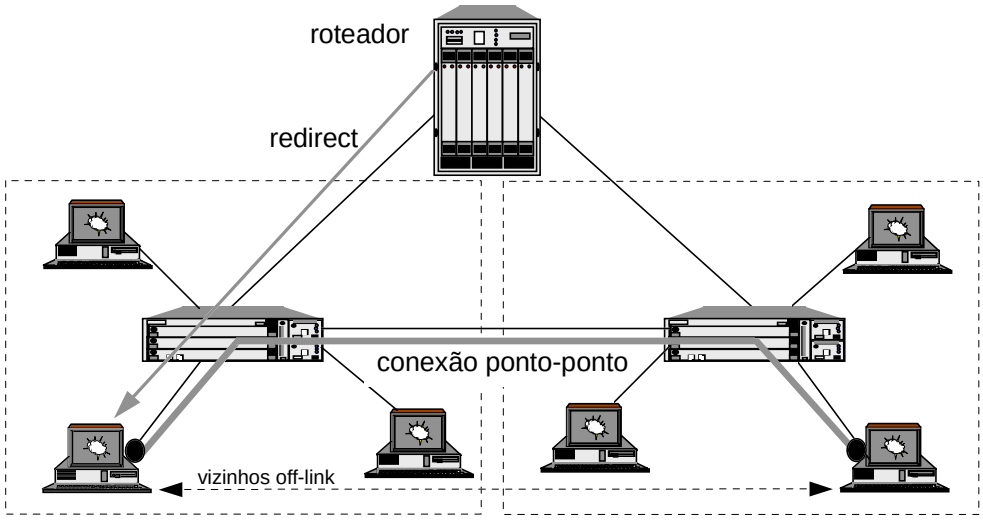


Figura 87: Comunicação entre hosts vizinhos em diferentes LLGs

A comunicação entre hosts não vizinhos se processa através de roteador (Figura 88).

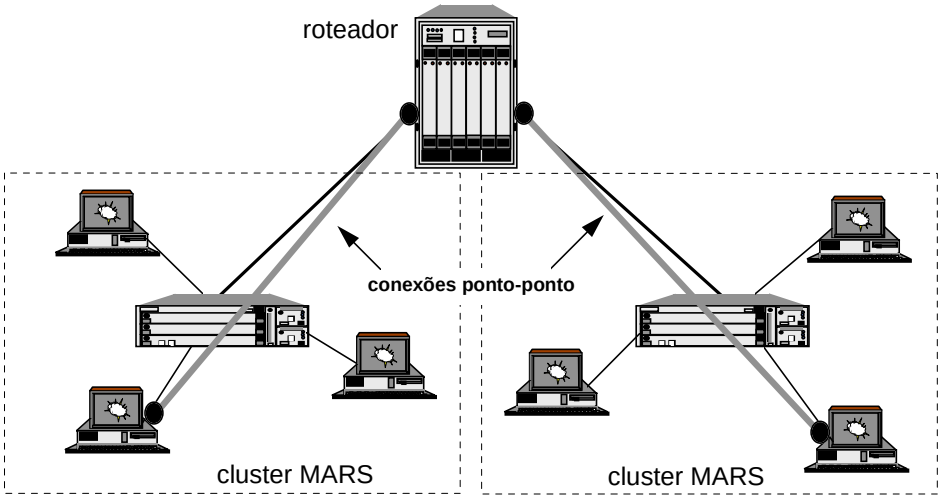


Figura 88: Comunicação entre hosts não-vizinhos.

Atualmente, existem 3 propostas de implementação do Neighbor Discovery sobre ATM: [Arm96b], [Sch96] e [Atk96] , entretando, o uso do MARS para suportar o caso genérico de multicast do IPv6 é independente da implementação do Neighbor Discovery.

Uma característica ausente do cenário descrito neste item é uma função de mapeamento de nomes para endereços e vice-versa que seja simples mas que leve em conta o caráter dinâmico dos endereços locais ao *link*.

[Har96] sugere que interfaces que suportem endereço local ao link anunciem tanto seu nome quanto seu endereço local ao link a um grupo de multicast com escopo no enlace local e que exista uma função de resolução de nome ou de endereço suportada no grupo de multicast que seria respondida pelo host identificado.

Tais mapeamentos de nome para endereço não deveriam ser tratados pelo DNS/BIND devido a seu escopo restrito ao enlace, não devendo ser propagados por nenhum roteador. Forçar o DNS/BIND a separar endereços de escopo local de endereços globais poderia ser um *overhead* prejudicial.

4.1.7 Suporte a Qualidade de Serviço

Os campos de *identificador de fluxo* e *prioridade* do cabeçalho do IPv6 podem ser usados por hosts para suportar aplicações que exijam alguma taxa de vazão consistente, do tipo associado a aplicações de tempo-real ou multimedia.

O campo de flow label é composto de 24 bits e, juntamente com o endereço de origem, identifica univocamente um fluxo. Um fluxo é uma seqüência de pacotes enviadas de uma certa fonte para um mesmo destino (que pode ser um endereço de unicast ou multicast).

O host de origem aloca um flow label pseudo aleatoriamente no valor entre 0x1 e 0xFFFFFF. O valor 0x0 é reservado para identificar ausência de suporte (ou de interesse) a fluxo.

Todos os pacotes pertencentes a um fluxo devem ser enviados com os mesmos endereços de origem e de destino, mesma prioridade e mesmo flow label. Caso exista cabeçalho de extensão hop-by-hop, este deve ser mantido ao longo do fluxo.

Roteadores podem estabelecer um controle de fluxo mesmo quando a origem não requisitar. Eles processam informações do tipo próximo *hop* e rota preferencial e podem armazenar estes valores em memória *cache*, usando o endereço de origem e o flow label como índice de forma a agilizar o processamento dos pacotes.

O campo *prioridade* contém 4 bits que permitem que a fonte estabeleça uma prioridade de entrega desejada de seus pacotes. O intervalo é dividido em 2 grupos, onde os valores de 0 a 7 são reservados a tráfego sujeito a controle de contenção (como o tráfego TCP) e os valores de 8 a 15 são reservados a tráfegos que não podem ser influenciados por contenções, como as aplicações multimedia e de tempo-real.

A inclusão do campo de flow label no cabeçalho do IPv6 teve em mente os protocolos de reserva de recursos, como o RSVP, segundo os autores, essenciais para suportar aplicações de tempo-real.

Para que aplicações suportem QoS, as seguintes informações deveriam ser mapeáveis (ou extraíveis) de pacotes de protocolos como IPv6:

- Endereço de Origem
- Endereço Destino
- Parâmetros de QoS da Conexão
- Estado da Conexão
- Identificador de Circuito Virtual ATM

sendo que algumas destas informações poderiam ser obtidas de protocolos de reserva de recursos. Entretanto, o identificador de circuito virtual ATM deveria ser derivável do pacote IPv6 ([Bra94]).
Nesse aspecto, existe pouco suporte do IPv6 ao ATM, no sentido de que fluxos não são circuitos virtuais e que rotas não são influenciadas pela utilização de fluxos. Não existe procedimento de setup de fluxos e não há, necessariamente, nenhum compromisso de alocação de banda ao se estabelecer um fluxo. Por outro lado, roteadores podem decidir alocar banda sob demanda em situações de congestionamento baseado em parâmetros estabelecidos pelos receptores através do RSVP.

4.1.8 Segurança

O IPv6 usa o conceito de associação de segurança para definir 2 formas de segurança nos pacotes: a extensão de autenticação e a carga segura criptografada. A primeira provê serviço de autenticação, no qual o recebedor do pacote garante que o endereço de origem é autêntico e que o pacote não foi alterado durante a transmissão. A outra garante que apenas os recebedores legítimos terão acesso ao conteúdo do pacote.

Header IPv6	Header de Autenticação	Header TCP + Dados

Header IPv6	Header de Roteamento	Header de Autenticação	Header TCP + Dados

Exemplos de pacotes TCP autenticados

«--- Não Encriptografado ---»			«--- Criptografado ---»
Header IPv6	Headers de Extensão	Header ESP	Dados Criptografados

Formato do pacote usando carga segura encriptografada (ESP - Encrypted Security Payload)

Praticamente não há diferença entre a filosofia de segurança do IPv4 e IPv6, já que foram propostas na mesma época. Contudo, o esforço de se implementar segurança nesses moldes no IPv4 atual é tão grande quanto atualizar para IPv6 e aproveitar as facilidades já disponíveis da nova versão.

4.1.9 Mecanismos de Transição do IPv4 Para o IPv6

Um dos pontos fundamentais da nova versão do IP foi projetar um mecanismo de transição o mais simples e fácil possível que permitisse a interoperabilidade de hosts IPv4 e IPv6 e a instalação gradativa e difusa de roteadores e hosts implementando IPv6. Nunca foi objetivo do IPv6 obsoletar o IPv4.

São características do mecanismo de transição do IPv6:

Instalação e atualização incrementais
Hosts e roteadores IPv4 podem gradual e independentemente ser atualizados para o IPv6.

Dependência mínima de atualizações
O único pré-requisito para atualização para o IPv6 é que o serviço de DNS seja atualizado primeiro para suportar endereços de 128 bits. No caso dos roteadores, não há pré-requisitos.

Endereçamento facilitado
Quando hosts e roteadores forem atualizados para IPv6 podem continuar usando seus endereços IPv4.

Baixos esforços iniciais
Pouco ou quase nenhum trabalho é exigido dos usuários e administradores de rede para atualizar sistemas IPv4 para Ipv6. A estrutura de endereços IPv6 pode embutir endereços IPv4 e codificar outras informações usadas pelos mecanismos de transição.
O modelo de instalação prevê que a primeira leva de hosts e roteadores a serem atualizados implementarão completamente os 2 stacks: o IPv4 e o IPv6. Esta solução, entretanto, apóia-se firmemente no DNS/BIND como forma de mapeamento dos endereços.

Foi criada uma técnica de encapsulação de IPv6 dentro de IPv4 (tunelamento) de forma a permitir a transmissão em segmentos que não suportem o Ipv6. Foi também criada a técnica de translação de cabeçalhos para permitir uma eventual introdução de topologias que roteiem apenas tráfego IPv6.

Tal como foram projetados os mecanismos de transição, o IPv6 pode interoperar com o IPv4 até que se esgotem os endereços IPv4. Mesmo após isto, ambos poderão continuar coexistindo indefinidamente, mas apenas num escopo restrito. Por exemplo, impressoras de rede jamais precisarão ser atualizadas para IPv6.

4.1.10 Comentários Finais

As especificações do IPv6 estão em sua grande maioria prontas e demonstram um admirável consenso por parte dos membros da IETF. Se por um lado ainda existem alguns membros e fabricantes exitantes em implementar esta solução, tentando desenvolver algo "melhor", por outro a nova versão do IPv6 se mostra bastante adequada para as características de uso que se prevê nos próximos anos.

Por outro lado, ainda não se definiu a implementação de alguns dos protocolos nativos do IPv6 sobre a tecnologia ATM, com propostas ainda em estudo. A própria utilização do ATM é questionada pelos autores do IPv6 devido à sua complexidade e ao baixo ganho que oferece. Não se trata meramente de ganho de banda. Muitos autores consideram que a tecnologia ATM tem tudo que impressiona a indústria de telecomunicações e exatamente por isso é totalmente antagônica à tecnologia de redes locais. Mesmo a questão de escalabilidade é contestada, pois segundo os autores a complexidade é tão grande que tornará inviável sua implantação.

Por outro lado, estes mesmos autores não descartam nem ignoram esta tecnologia e assumem que a integração do IP sobre ATM não é tão mais complicada que as implementações de IP sobre X.25 existentes hoje, pelo menos quando se trata de circuitos permanentes. Ao mesmo tempo, apostam no uso apenas restrito do ATM na Internet, sugerindo soluções com menos *overhead*, como implementar o IPv6 sobre Sonet, utilizar o Fast Ethernet nas redes locais e apontam outras tecnologias com banda larga mais voltadas para a realidade das redes locais (como HIPPI, Fiber-Channel ou Myrinet).

Um ponto destacado por [Hui96] é a de que a qualidade de serviço do ATM foi desenvolvida há cerca de 15 anos, quando as necessidades de aplicações de tempo-real e multimedia eram outras; hoje, o que realmente seria necessário para se suportar multimedia na Internet é largura de banda.

O ATM destaca-se, então, quando se busca uma integração das redes locais com a indústria das telecomunicações, mas seu uso em larga escala é ainda incerto até que se concluam as especificações dos protocolos mais importantes e que testes sejam feitos em larga escala.

4.2 RSVP (Resource reSerVation Protocol) Sobre ATM[21]

O protocolo RSVP é usado por um host para especificar uma determinada qualidade de serviço (QoS) para um fluxo na rede. Esse protocolo é usado para enviar uma requisição de qualidade de serviço para todos os nós ao longo do caminho de transmissão a fim de que os recursos necessários sejam reservados ao longo do caminho. Esta requisição é propagada no sentido contrário ao fluxo, do receptor para o emissor (Figura 89).

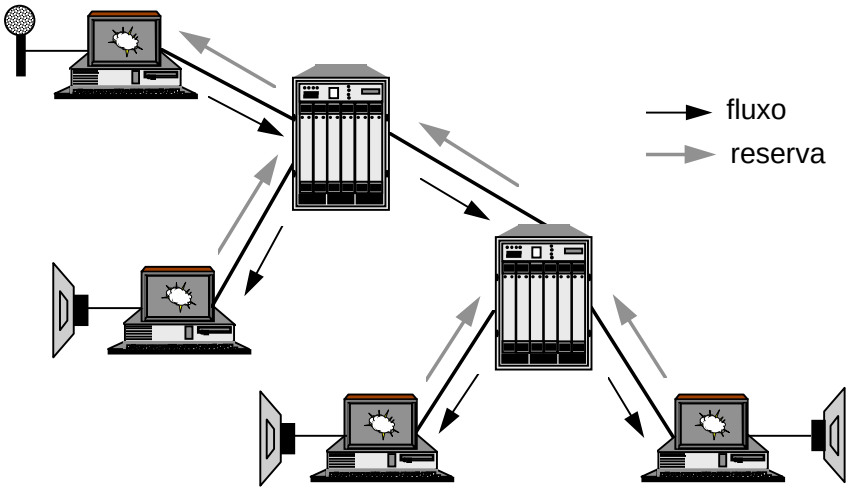


Figura 89: Esquema de reserva no RSVP.

Um aplicativo que deseja obter um determinado nível de QoS utiliza uma interface (API - *Application User Interface*) para comunicar sua requisição ao RSVP (Figura 90). Uma vez comunicada, o RSVP passa a requisição ao módulo de controle de políticas para verificação de permissão e, caso suceda, o controde de admissão decidirá pela aceitação do fluxo com base nas estimativas de recursos utilizados. Uma vez aceito o fluxo de pacotes, o *daemon* RSVP atua sobre os módulos de classificação e escalonamento de pacotes e modifica a base de controle de tráfego para que este novo fluxo seja corretamente tratado. Finalmente, o RSVP passa imediatamente a requisição para o próximo nó da rede, definido pela interface de roteamento local.

A partir deste momento todo pacote do novo fluxo que chegar será enviado ao mecanismo de classificação que vai associá-lo a uma classe. O mecanismo de escalonamento então controla o envio dos pacotes, reorganizando ou até mesmo descartando os pacotes em suas classes, respeitando os compromissos de QoS assumidos com os aplicativos.

[21] Baseado na referência [Pag96]

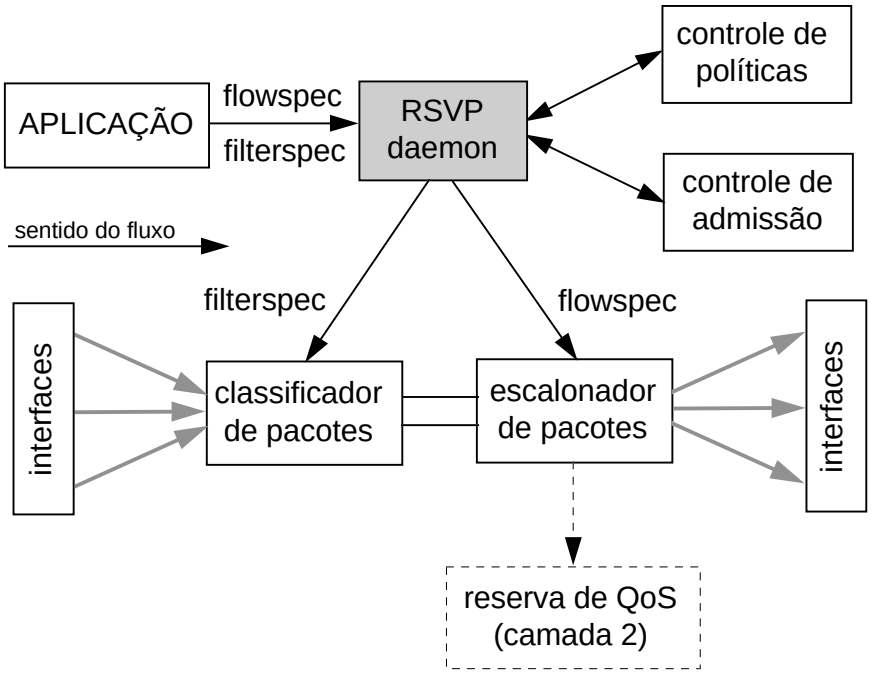


Figura 90: Arquitetura do RSVP

4.2.1 Características Gerais

RSVP é um protocolo simplex, isto é, realiza reservas de recursos em apenas uma direção, tanto para transmissões *unicast* como para *multicast*. Essas reservas devem ser feitas pelo receptor no sentido inverso do tráfego das mensagens, conferindo ao RSVP a característica de ser orientado ao receptor.

RSVP não é um protocolo de roteamento, apenas interage com os protocolos existentes para determinação de rotas para envio das mensagens necessárias.

RSVP mantém em todas as máquinas do caminho informações de maneira *soft-state*, ou seja, todas as informações são atualizadas periodicamente pelas mensagens do fluxo. Caso contrário as informações são automaticamente descartadas e as reservas desativadas.

A característica *soft-state* permite a mudança dinâmica no conjunto de fontes ou requisições de QoS através de simples mudança nos parâmetros de QoS e tráfego. Além disso o estabelecimento de rota é dinâmico e também atualizado por mensagens periódicas. Deste modo, as falhas são tratadas automaticamente através de desativação da reserva por *timeout*. Além da desativação automática, o RSVP também permite a desativação explícita .

Outra característica do RSVP é a agregação de mensagens de reserva que permite evitar que tráfego duplicado congestione a rede. Se mais de uma requisição tiver de ser enviada pela rede para o mesmo conjunto de fonte, elas serão agregadas em uma única requisição contento a maior qualidade de serviço especificada.

Um conceito importante definido pelo RSVP é o conceito de sessão. Sessão é um fluxo de dados com endereço destino, identificador de protocolo e opcionalmente uma porta destino. Através dos campos de identificação da sessão, os receptores poderão escolher determinados tipos de pacotes.

4.2.2 Modelo de Reservas

RSVP define diversos mecanismos para caracterização de uma reserva de qualidade de serviço para um fluxo de dados. As principais definições são quanto aos tipos de mensagens usadas para criação e estabelecimento de reservas de um fluxo, e quanto aos diversos estilos definidos para melhor adaptação da necessidade dos aplicativos às características de transmissão da sessão. Na sequência serão abordadas essas definições e serão apresentados alguns exemplos de reserva ilustrando algumas características do RSVP.

Tipos de mensagens

O transporte de mensagens realizado pelo RSVP é feito de maneira opaca, isto é, a definição do protocolo não se preocupa com o tipo nem o formato das mensagens utilizadas. Deste modo novas mensagens podem ser definidas no futuro para adequação à novas necessidades.

As principais mensagens transportadas pelo RSVP são: mensagens caminho (PATH) e mensagens reserva (RESV) que serão descritas em mais detalhes a seguir.

Mensagens RESV

Conhecidas como “descritoras de fluxo” modelam o tráfego e qualidade de serviço desejados, além de especificar filtros nos quais determinados pacotes de uma sessão podem ser escolhidos. Mensagens RESV são formadas dos seguintes componentes:

FlowSpec - caracteriza a QoS e o tráfego definidos pelos seguintes parâmetros:

- Modelo de Serviço - Garantido, Preditivo, Carga Controlada.
- Tspec (Caracterização de tráfego) - descreve o tráfego do fluxo em parâmetros *token-bucket*:

- **r** (token bucket rate) - banda sustentada (bytes/s) pelo emissor;
- **b** (token bucket size) - violação permitida da banda sustentada por um período T (bytes);
- **p** (peak data rate) - banda de pico do emissor (bytes/s);
- **m** (minimum policed unit) - menor pacote gerado pelo emissor (bytes);
- **M** (maximum packet size) - maior pacote gerado pelo emissor (bytes);
- **R** (rate) - banda solicitada (bytes/s);
- **S** (slack term) - atraso solicitado (s).

- Rspec (Caracterização de reserva) - define a QoS desejada através de parâmetros específicos de cada modelo:
- Serviço Carga Controlada - garante os parâmetros **r**, **b**, **p**, **m**, **M** da especificação do fluxo;
- Serviço Garantido - garante os parâmetros **r**, **b**, **p**, **m**, **M**, **R**, **S** da especificação do fluxo.
- Serviço de Melhor Esforço - nenhum parâmetro é garantido (este serviço é o único disponível face a presença de roteadores que não suportam RSVP na rota).

FilterSpec - escolhe pacotes a serem recebidos, especificando alguns campos do identificador de sessão:
- IPv4: endereço IP + TCP/UDP port da origem;
- IPv6: identificador de fluxo (ou filterspec do IPv4).

A Figura 91 ilustra o tráfego de mensages RESV.

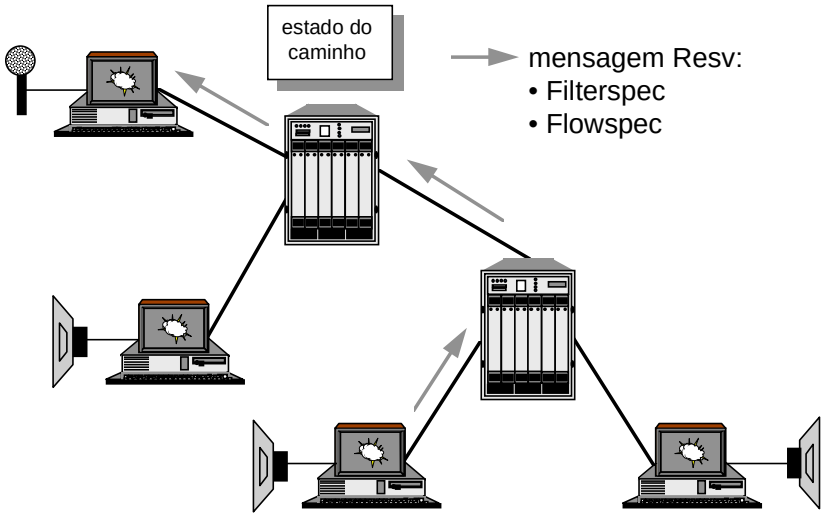


Figura 91: Fluxo de mensagens RESV.

Mensagens PATH

Uma mensagem PATH estabelece o caminho com configurações de reserva para transmissão dos dados do fluxo. Utiliza um campo denominado *last-hop* armazenado em cada máquina ao longo do caminho, até o receptor, com o objetivo de encaminhar as mensagens RESV pelo mesmo caminho, porém em sentido inverso. A mensagem PATH também pode transportar informações sobre o tráfego gerado e as condições da rede para que os receptores possam montar requisições mais susceptíveis de serem aceitas. Os principais componentes são descritos abaixo:

* *Sender Template* - descreve a forma dos pacotes de dados originados pela fonte de transmissão, que será utilizado junto com o *filterspec* para a seleção de pacotes.
* *Sender Tspec* - descreve o tráfego gerado pela fonte em parâmetros *token-bucket*, que será enviado para o receptor e roteadores ao longo da rede para informar as características de tráfego e que tipo de reserva esperar.
* *Adspec* - mensagem que implementa o modelo de reserva OPWA (*One-Pass With Advertising*), que colhe informações de todos os roteadores do caminho e informa ao receptor o estado da rede, para construção de reservas mais refinadas. Mensagens Adspec informam:
* a preseça de roteadores que não suportam RSVP no caminho;
* a banda estimada do caminho (bytes/s);
* a latência (atraso) mínima do caminho (s);
* o MTU do caminho;

A Figura 92 ilustra o fluxo de mensagens PATH.

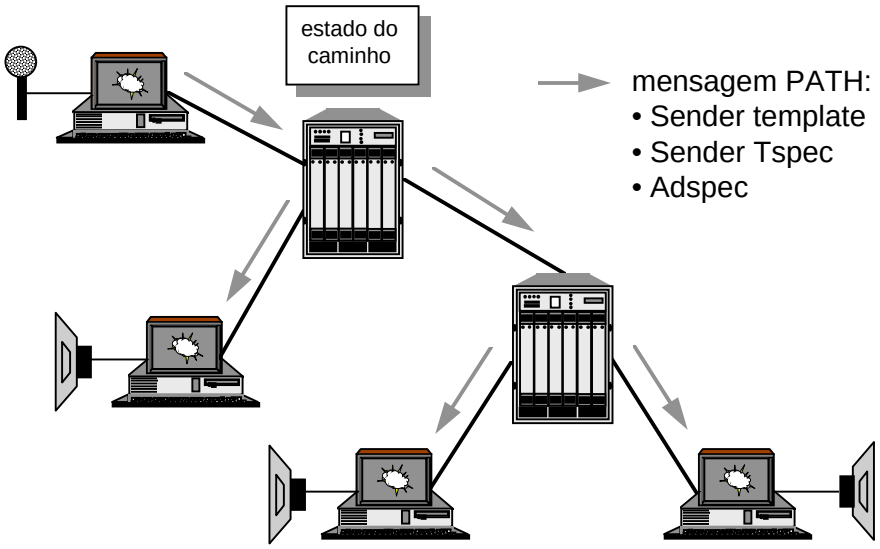


Figura 92: Fluxo de mensagens PATH.

4.2.3 Estilos de Reserva

Uma requisição de reserva inclui um conjunto de opções que são conhecidas como estilos de reservas. As duas principais opções são relativas ao modo de estabelecimento do uso do canal de transmissão e à definição da seleção das fontes que serão aceitas na sessão.

A opção referente ao uso do canal determina se as reservas serão realizadas de forma exclusiva para cada fonte escolhida ou se o canal será compartilhado por todas as fontes da sessão. A segunda opção estabelece como será a escolha das fontes de transmissão para uma determinada sessão. Esta escolha pode ser feita entre a determinação explícita de todas as fontes (utilizando as mensagens de *filter spec*) ou implícita, na qual todas as fontes da sessão serão selecionadas. A tabela a mostra o relacionamento das opções de reserva com os respectivos modelos que serão definidos a seguir:

	Reservas	
Seleção da fonte	Exclusivo	Compartilhado
Explícita	Estilo *Fixed-Filter* (FF)	Estilo *Shared-Explicit* (SE)
Implícita	Não definido	Estilo *Wildcard-Filter* (WF)

Tabela A: Estilos e Atributos de Reserva do RSVP.

- Estilo WF - cria um único canal compartilhado por todas as fontes da sessão. WF(*{Q}) significa um receptor implícito (*, todo) e uma qualidade de serviço (Q) compartilhada. A **Error! Reference source not found.**Figura 93 ilustra um exemplo de reserva WF.

- Estilo FF - cria um canal exclusivo para cada fonte explicitamente escolhida. FF(S1{Q1},S2{Q2}, ...) significa receptores explicitamente escolhidos (S1,S2,...) e seus respectivos níveis de QoS (Q1, Q2). A Figura 94 ilustra um exemplo de reserva FF.

- Estilo SE - cria um único canal compartilhado por cada fonte explicitamente escolhida. SE((S1,S2,...){Q}) significa receptores explicitamente escolhidos (S1,S2,...) e uma qualidade de serviço (Q) compartilhada. A Figura 95**Error! Reference source not found.** ilustra um exemplo de reserva SE.

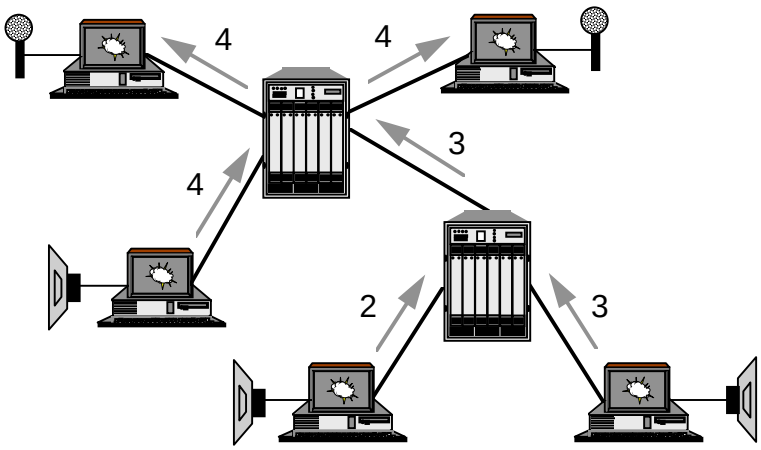


Figura 93: Exemplo de reserva WF.

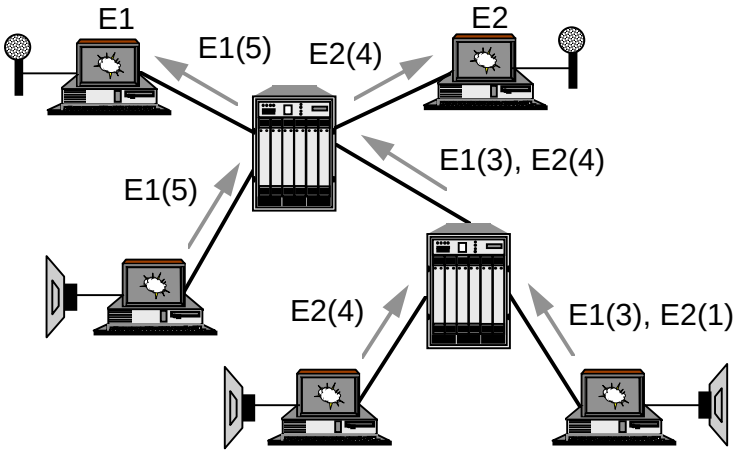


Figura 94: Exemplo de reserva FF.

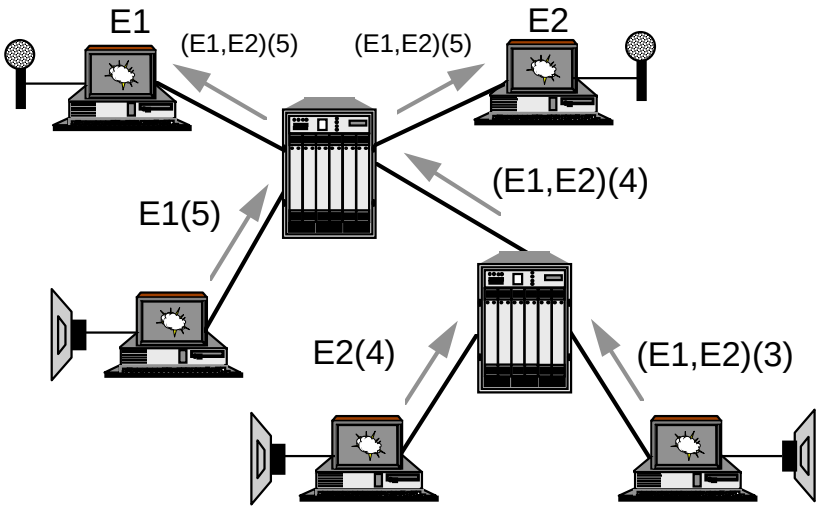


Figura 95: Exemplo de reserva SE.

4.2.4 Criação de uma Sessão RSVP

Antes que uma sessão RSVP seja criada, o identificador de sessão precisa ser atribuído e comunicado para todos os receptores e fontes por algum tipo de mecanismo não definido pelo modelo se Serviço Integrado (IS). Depois de distribuído o identificador os seguintes passos (Figura 96) são necessários para criação da sessão:

1. Algum receptor faz a requisição de entrada no grupo *multicast* daquela sessão através de uma mensagem IGMP (*Internet Group Management Protocol*).
2. O roteador *multicast* que receber a requisição de inserção no grupo do receptor vai inundar a rede de mensagens para a construção de rotas em todas as máquinas até a fonte de transmissão.
3. Recebido o aviso de que existe pelo menos um receptor, a fonte começa enviar mensagens PATH para este receptor, construindo o caminho no qual serão realizadas reservas para que os pacotes de dados possam fluir com qualidade de serviço. Essas mensagens PATH utilizam o campo *last-hop* para indicar ao próximo nó do caminho como rotear pelo mesmo caminho, porém no sentido inverso, as mensagens de reserva.
4. O receptor recebe a primeira mensagem PATH informando o estado do caminho para construção das mensagens de reserva.
5. O receptor envia as mensagens RESV pelo mesmo caminho que foi construído pelas mensagens PATH através da utilização da variável *last-hop* armazenada em cada nó do caminho.
6. Cada roteador recebe a mensagem RESV, submetendo-a ao módulo de controle de admissão, e em caso de sucesso realiza as reservas necessárias e modifica a base de

controle de tráfego. No caso de falha envia uma mensagem de erro para o receptor de origem. A mensagem RESV é então recebida pela fonte de transmissão.

7. A fonte então começa a enviar os dados com a qualidade de serviço requisitada.

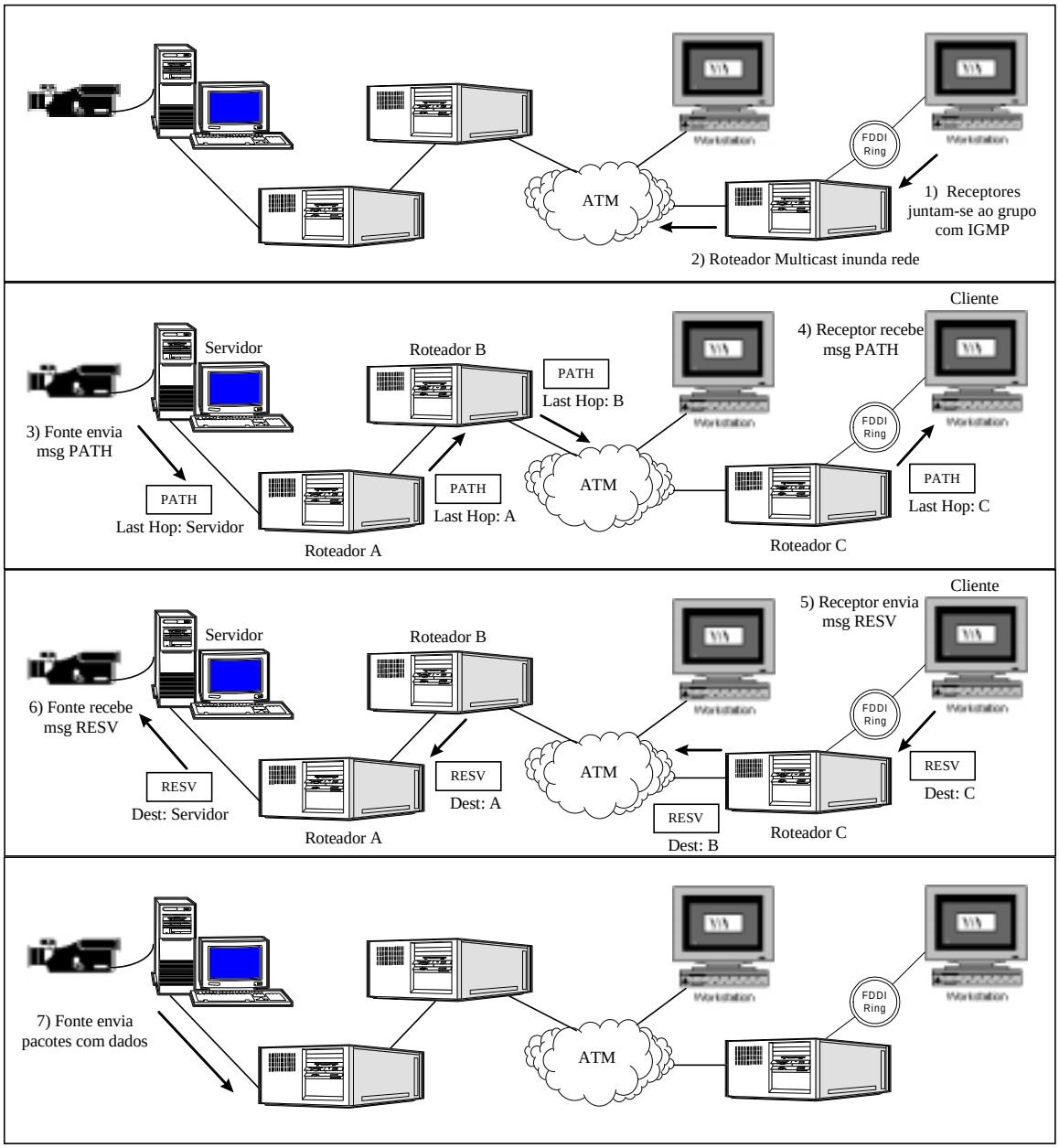


Figura 96: Abertura de uma sessão RSVP.

Note que, para o caso de adição de um receptor onde já existem reservas para outros receptores, os mesmos passos são seguidos com uma pequena diferença, os

dados começam a fluir no momento em que o receptor junta-se ao grupo *multicast* ainda sem qualidade de serviço, pois as mensagens de reserva não foram enviadas ainda.

4.2.5 RSVP sobre ATM

Redes ATM usam o protocolo de sinalização Q.2931 não só para estabelecimento de conexões mas também para alocar recursos para as conexões. O Q.2931 tem muitas características de um protocolo de sinalização convencional, tais como ser orientado à origem e manter informações permanentes nas chaves para manter as conexões (*hard-state*). Na especificação 3.0/3.1 da UNI a QoS associada com a conexão no seu estabelecimento não pode ser mais modificada (ou seja, é estática). Numa conexão unicast recursos são alocados em ambas as direções ao longo do caminho, enquanto no caso multicast, eles são alocados somente da origem para o destino. Nesta versão todos os destinatários recebem a mesma QoS.

O protocolo IP provê um serviço não orientado à conexão. Redes ATM, por outro lado, proporcionam um serviço orientado à conexão, onde os recursos são reservados na inicialização da conexão, usando a interface de rede do usuário (UNI) e um protocolo de sinalização (NNI). A tabela 5 ilustra estas diferenças.

Característica	**RSVP**	**ATM (UNI 3.x)**
Orientação	Receptor	Fonte
Conexão	*Soft-State* (*refresh time-out*)	*Hard-State* (explícito)
Configuração de QoS	Separado da definição rota	Momento de definir rota
Mudanças de QoS	Dinâmicas	Estáticas (configuração)
Direção de reservas	Simplex	Duplex unicast Simplex multicast
Heterogeneidade de QoS	Sem limite	Uniforme

Tabela 5: Comparação do RSVP e ATM UNI 3.x.

Os princípios usados no projeto do RSVP diferem daqueles do ATM nos pontos abaixo:

- As reservas de recursos são representadas por informações que são periodicamente renovadas em um determinado período (*soft-state*), isto é, reservas não são permanentes. Se as informações não são atualizadas elas são explicitamente removidas. Em ATM, recursos são reservados por toda a duração da conexão, que precisam ser explicitamente removidas de maneira confiável.
- A solução não orientada a conexão do RSVP permite que as reservas de QoS de um fluxo possam ser alteradas a qualquer momento, enquanto conexões ATM tem QoS estático que são negociadas no estabelecimento da conexão.
- RSVP é um protocolo simplex, isto é, os recursos são reservados em uma direção somente. Em ATM, conexões (e reservas associadas) são bidirecionais em conexões ponto-a-ponto e unidirecionais em conexões ponto-multiponto.
- A reserva de recursos é iniciada pelos receptores no RSVP. Em ATM, os recursos são reservados pelo sistema que inicializa a conexão. Nas conexões ponto-multiponto, a inicialização da conexão (e consequentemente reserva de recursos) precisa ser feita pela origem.
- RSVP tem suporte para sessões contento múltiplas fontes, e para chavear dinamicamente entre fontes. Nenhum destes suportes são proporcionados pelo ATM.
- RSVP foi projetado independentemente de outros componentes da arquitetura, em particular o roteamento. Além do mais, a determinação da rota e reserva de recursos são feitos em tempos diferentes. No ATM, reserva de recursos e determinação de rotas são feitas ao mesmo tempo (da inicialização da conexão).

Devido a característica orientada à conexão do ATM os roteadores RSVP necessitam realizar a gerência da abertura e do fechamento das conexões ATM quando reservas RSVP são feitas ou liberadas (por *time-out*). O melhor esquema para o gerenciamento de conexões depende do custo de manter a conexão aberta para um futuro uso por outro fluxo. Por exemplo, conexões abertas com QoS podem ser utilizadas para tráfego melhor esforço (desde que exista disponibilidade de banda) Por outro lado, uma conexão com QoS é cara desde que os recursos necessários precisam ficar reservados.

Outra característica em que o RSVP apresenta conflito com o ATM é no uso de mensagens caminho para transportar informações para os receptores antes que a reserva seja feita. Deste modo a reserva de recursos se dá em separado do roteamento. A entrega de mensagens PATH através de uma rede ATM requer um mecanismo de inicialização de conexões sem reservas. A conexão necessita ser razoavelmente confiável para que pelo menos algumas mensagens caminhos sejam entregues.

Atualmente, RSVP sobre ATM se limita a mapear os serviços RSVP (Garantido, Carga Controlada e Melhos Esforço) em tipos de conexões ATM (CBR, ABR, etc). A Figura 97 ilustra o mapeamento sugerido. Ao abrir-se uma conexão ATM para o

fluxo RSVP, os parâmetros Tspec devem ser mapeados em parâmetros de QoS ATM. Por exemplo:

Parâmetros RSVP (Tspec):
r (token bucket rate) - banda sustentada (bytes/s) pelo emissor
b (token bucket size) - violação permitida da banda sustentada por um período T (bytes)
p (peak data rate) - banda de pico do emissor (bytes/s)

Parâmetros ATM:
SCR (sustainable cell rate) - taxa de células sustentada num VC
PCR (peak cell rate) - taxa máxima de transmissão num VC
MBS (maximum burst size) - número de células transmitidas num pico

Mapeamento[22]:
SCR = r / 48
MBS = b
PCR = p / 48

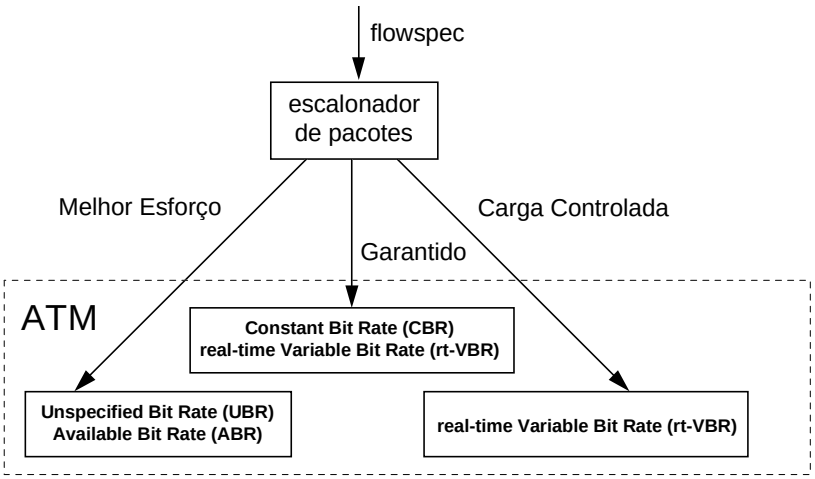


Figura 97: Mapeamento dos tipos de serviço RSVP em conexões ATM (UNI 3.x).

[22] Desconsiderando-se o overhead de segmentação e remontagem.

5 Bibliografia

[ACM95] Communications of the ACM - volume 38, #2, Fevereiro 1995

[Alb96] Alberti, A.M; Lima, M.T.G.; Cezar, P.G. – "IP sobre ATM", Monografia apresentada no Curso de IA365 Tópicos em Eng. de Computação – 1996, FEEC/UNICAMP;

[All95] Alles, Anthony - "ATM Internetworking", Cisco Systems, Maio 1995

[Arm96a] Armitage, Grenville - "IPv6 and Neighbor Discovery over ATM", Internet Draft, Junho 1996

[Arm96b] Armitage, Grenville - "Transient Neighbors for IPv6 over ATM", Internet Draft, Junho 1996

[Atk96] Atkinson R.; Haskin D.; Luciani J. - "IPv6 over NBMA Networks", Internet Draft, Novembro 1996

[Atm95] ATM-Fórum – LAN Emulation;

[Atm97] ATM-Fórum - MPOA;

[Bra94] Bradner, B.; Mankin, A. - "The Recommendation for the IP Next Generation", RFC 1752, Janeiro 1994

[Brz94] Brazdziunas, Cristina - "IPng Support for ATM Services", RFC 1680, Agosto 1994

[Con96] Conta, A.; Deering, S. - "Internet Control Message Protocol, version 6 Specification", RFC 1885, janeiro 1995

[Dee95a] Deering, S.; Hinden, R. - "Internet Protocol, Version 6 (IPv6) Specification", RFC 1883, dezembro 1995

[Dee95b] Deering, S.; Hinden, R. - "IP Version 6 Addressing Architecture", RFC 1884, março 1995

[Dor96] Dorling; Freedman; Metz – "Internetworking with ATM", Prentice-Hall 1996;

[Gin96] Ginsburt, David – "ATM Solutions for Enterprise Internetworking", Addison-Wesley, 1996

[Har96] Harrington, D. - “Link Local Addressing and Name Resolution in IPv6”, Internet Draft, Janeiro 1996

[Hin96] Hinden, Robert - “IP Next Generation Overview” Communications of the ACM - volume 39, #6, Junho 1996

[Hui94] Huitema, Christian - “The H Ratio for Address Assignment Efficiency”, RFC 1715, Novembro 1994

[Hui96] Huitema, Christian - “IPv6: The New Internet Protocol”, Prentice Hall, 1996

[McD94] McDysan ; Spohn - “ATM Theory and Application”, McGrall Hill, 1994

[Min96] Minoli; Alles A. – “LAN, ATM, and LAN Emulation Technologies, Artech House, 1996

[New96] Newman;P. et.all – “IP Switching: ATM under IP, White Paper, Ipsilon Networks, 1996;

[Oli96] Oliveira, A.M.; Oliveira, J.C.; Gome, L.C.T. – “LAN Emulation”, Monografia apresentada no Curso IA 364, 1996, FEEC/UNICAMP;

[Pag96] Pagani, C. – “RSVP sobre ATM” – Monografia apresentada no Curso IA 364, 1996, FEEC/UNICAMP;

[RFC] RFCs: 1577; 1483; 1209; 1626; 826; 1755; 1970; 1954; 1986; 1953; 1883;

[Sac96] Sackett, G.C.; Metz, C.Y. – “ATM and Multiprotocol Networking, McGraw-Hill, 1996

[Sch96] Schulter, P. - “A Framework for IPv6 over ATM”, Internet Draft, Fevereiro 1996

[Wad96] Wada, W. – “IPv6 sobre ATM”- Monografia apresentada no Curso IA 364, 1996, FEEC/UNICAMP.