

**Fernando Antônio Cerutti**

**Implementação de um Ambiente de Gerência em Redes  
ATM Utilizando a Tecnologia WEB**

**Florianópolis  
1999**

**UNIVERSIDADE FEDERAL DE SANTA CATARINA  
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIAS DA  
COMPUTAÇÃO**

**Implementação de um Ambiente de Gerência em Redes  
ATM Utilizando a Tecnologia WEB**

Dissertação submetida à Universidade Federal de Santa  
Catarina como parte dos requisitos para a obtenção do  
grau de Mestre em Ciências da Computação

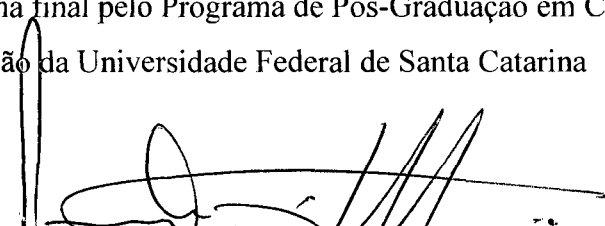
**FERNANDO ANTONIO CERUTTI**

Florianópolis, Agosto de 1999.

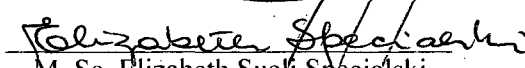
# IMPLEMENTAÇÃO DE UM AMBIENTE DE GERÊNCIA EM REDES ATM UTILIZANDO A TECNOLOGIA WEB

Fernando Antonio Cerutti

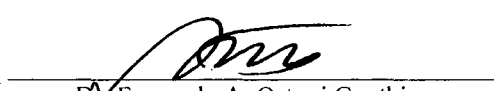
Esta Dissertação foi julgada adequada para obtenção do Título de Mestre em Ciências da Computação. Área de concentração Sistemas Computacionais, e aprovada em sua forma final pelo Programa de Pós-Graduação em Ciências da Computação da Universidade Federal de Santa Catarina



Dr. Paulo José de Freitas Filho  
Orientador

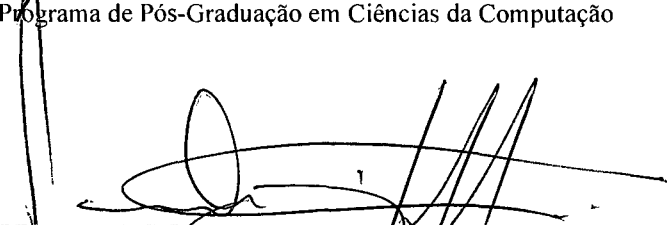


M. Sc. Elizabeth Sueli Specialski  
Co-Orientadora

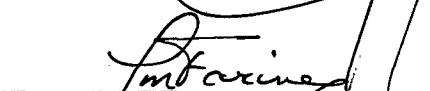


Dr. Fernando A. Ostuni Gauthier  
Coordenador do Programa de Pós-Graduação em Ciências da Computação

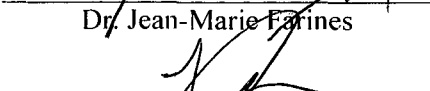
Banca Examinadora:



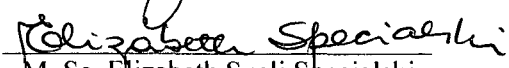
Dr. Paulo José de Freitas Filho  
Presidente



Dr. Jean-Marie Farines



Dr. João Bosco da Mota Alves



M. Sc. Elizabeth Sueli Specialski

## Resumo

A proposta deste trabalho foi implementar um ambiente de gerência para redes ATM (*Asynchronous Transfer Mode*) integrado com a tecnologia Web. Para essa finalidade, elaborou-se um estudo detalhado das variáveis das MIBs (*Management Information Base*) de gerência relativas a parâmetros de tráfego e conexões virtuais das redes ATM. Tais variáveis foram relacionadas com as questões teóricas de tráfego, estabelecimento de conexões e a monitoração de alguns recursos vitais para o funcionamento dessas redes.

Através deste estudo implementou-se um ambiente de gerência que pode ser utilizado tanto pelo gerente da rede, com a finalidade de controlar a utilização de recursos, quanto pelos usuários finais, com a finalidade de obter históricos sobre essa utilização. Tal facilidade é oferecida pela implementação em código Java, que tornou possível que a ferramenta seja utilizada a partir de qualquer ponto da Internet que possua um browser Web.

O ambiente implementado permite controle e/ou monitoração de vários parâmetros relativos ao tráfego e aos recursos da rede, tais como: tráfego nas portas físicas, nos links virtuais, banda alocada em cada porta, endereços ATM dos equipamentos entre os quais foram estabelecidas as conexões.

## Abstract

The propose of this work was to implement a management environment for ATM (*Asynchronous Transfer Mode*) networks integrated with the Web technology. To reach this goal, a MIB (*Management Information Base*) study was elaborated. The analysed variables are related with traffic and virtual connections in ATM. These variables was correlated with conceptual questions about traffic, virtual connections and resource monitoring in ATM networks.

By means of this study a management environment was developed. This tool can be handled both by network managers and users. The managers may intend to control and monitoring the utilization of network resources. The final users can verify the amount of shared resources that are being utilized at any time. These facilities only may be possible with the Java code implementation. In this way, any point at Internet with a Browser may access the management environment.

The implemented environment have control over many network traffic and resources parameters, like physical interface traffic, virtual link traffic, alocated bandwidth at any interface and ATM addresses of devices conected through a virtual channel.

## Agradecimentos

Este estudo jamais poderia ter sido concluído sem a ajuda de um grande número de pessoas, e de pessoas amigas destas pessoas, de forma que fica impossível enumerá-las. Mas ficou demonstrado que a grande teia social que nos envolve, cheia de sentimentos, afinidades, emoções e incentivos, às vezes quase imperceptíveis, deve ser considerada sempre em lugar de destaque, primeiríssimo plano, abaixo da camada física.

De alguma forma, algumas pessoas envolveram-se mais diretamente nesse projeto. Fica um agradecimento especial a elas:

Edison Tadeu Lopes Melo, nosso (*Very*) “*Big Boss*” e mentor intelectual.

Jussara Maria Bozzano, André Melo Barotto, Elvis Melo Vieira, Adriano Souza e Kathia Jucá, pelo acompanhamento e companheirismo. Com vocês, as redes são bem mais simples, divertidas, esotéricas e não protocolares.

Alessandro Lemos, nosso *Java-man* e Kátyra Kowalski Armanini, nossa *Java-Girl*.

Solange Terezinha Sari, pela amizade e auxílio na revisão.

Márcio Cledes, Diretor do Núcleo de Processamento de Dados da UFSC, e nosso “*Database-resolver*”.

Prof. Paulo Freitas, pela orientação.

Prof.a. Elisabeth Speciaslki, pelo constante incentivo.

Prof. Jean-Marie Farines, pelo exemplo profissional.

Prof. João Bosco da Mota Alves, pela participação na banca examinadora.

Claudete Schilling Mendonça, que me ensinou a segurar a primavera nos dentes durante as maiores tempestades. Além disso, responsabilizou-se pelo padrão gráfico das figuras.

A equipe do NPD-UFSC, e em especial ao pessoal do Laboratório de Interoperabilidade de Redes, bolsistas, estagiários, voluntários, sonhadores de bits, links e protocolos.

Ao Rock Progressivo Britânico e a *New Age Music*, que fizeram a trilha sonora.

## **Dedicatória**

**Para meu pai, Avelino;  
minha mãe, Ilda;  
minha esposa, Claudete,  
e minha filha, Jéssica.**

# Índice

<b>1. Introdução</b> .....	<b>1</b>
<b>1.1. Descrição do problema</b> .....	<b>2</b>
<b>1.2. Descrição da proposta</b> .....	<b>3</b>
<b>2. Objetivos</b> .....	<b>5</b>
<b>2.1. Objetivos Gerais</b> .....	<b>5</b>
<b>2.2. Objetivos Específicos</b> .....	<b>5</b>
2.2.1. Implantação da rede.....	5
2.2.2. Monitoração do estabelecimento das conexões virtuais (Rotas Virtuais-VPs e Canais Virtuais-VCs). ....	6
2.2.3. Identificação dos Níveis de QoS negociados em cada Conexão.....	6
2.2.4. Estudo das variáveis das MIBs.....	6
2.2.5. Identificação dos níveis críticos dos parâmetros.....	6
2.2.6. Especificar e implementar o ambiente de gerência <i>ATRM-WTool</i> (ATM Traffic and Resources Management Web Tool).....	6
<b>3. O ambiente de estudos</b> .....	<b>8</b>
<b>3.1. A estrutura ATM na UFSC</b> .....	<b>8</b>
<b>3.2. Descrição do Ambiente de Estudos</b> .....	<b>9</b>
<b>4. Introdução à Tecnologia ATM</b> .....	<b>11</b>
<b>4.1. O que é o ATM</b> .....	<b>11</b>
<b>4.2. Objetivos do ATM</b> .....	<b>12</b>
<b>4.3. Topologia das redes ATM</b> .....	<b>13</b>
<b>4.4. VPI e VCI</b> .....	<b>15</b>
<b>4.5. Comutadores (switches)</b> .....	<b>15</b>
<b>4.6. Sinalização nas redes ATM</b> .....	<b>16</b>
<b>4.7. Controle de Admissão de Conexão – CAC</b> .....	<b>19</b>
Serviço de SVC.....	19
<b>5. Gerência de Redes</b> .....	<b>25</b>
<b>5.1. Modelo funcional OSI de Gerência de redes</b> .....	<b>25</b>
<b>5.2. Padronização de Gerência</b> .....	<b>25</b>
<b>5.3. SNMP</b> .....	<b>27</b>
<b>5.4. Elementos da Arquitetura de Gerência de Redes SNMP</b> .....	<b>27</b>
<b>5.5. Gerência na WEB</b> .....	<b>33</b>
5.5.1. Novas Tecnologias de Gerência.....	33
5.5.2. Gerência de redes na Web.....	34
5.5.3. Os Benefícios da gerência baseada na Web.....	35
5.5.4. Estratégias de implementação:.....	36
<b>5.6. Java e SNMP</b> .....	<b>38</b>
<b>6. Gerência de Redes ATM</b> .....	<b>41</b>
<b>6.1. Módulos de Gerência ATM</b> .....	<b>44</b>
6.1.1. MIBs das categorias M1 e M2.....	44
6.1.2. MIBs da categoria M3.....	45



6.1.3.	MIBs da interface M4:.....	45
6.1.4.	Outras MIBs: .....	46
6.1.5.	MIBs do IETF (Internet Engineering Task Force).....	46
6.1.6.	MIBs de outros órgãos de padronização .....	47
<b>6.2.</b>	<b>Gerência das Conexões ATM .....</b>	<b>48</b>
6.2.1.	Gerência das Conexões Virtuais Permanentes.....	48
<b>6.3.</b>	<b>Gerência de Recursos ATM .....</b>	<b>51</b>
6.3.1.	Contrato de Tráfego.....	51
6.3.2.	A integridade de serviço .....	51
6.3.3.	Contenção .....	52
6.3.4.	Funções de QoS na Gerência de Tráfego ATM.....	53
6.3.5.	Funções Genéricas de Gerência de Tráfego.....	53
6.3.6.	Arquitetura dos serviços ATM .....	56
6.3.7.	Definições das Categorias de Serviços .....	57
6.3.8.	Definições dos parâmetros de Qualidade de Serviço.....	58
<b>6.4.</b>	<b>Requerimentos para controle de tráfego e congestionamento em redes ATM.....</b>	<b>61</b>
6.4.1.	Efeitos da latência/velocidade .....	62
6.4.2.	Cell Delay Variation .....	64
6.4.3.	<i>Internal Queuing and Transmission Delay:</i> .....	67
6.4.4.	<i>External Queuing and Transmission Delay:</i> .....	68
6.4.5.	<i>Propagation Delay:</i> .....	68
6.4.6.	Processing Delay .....	68
6.4.7.	Fontes de Retardos Internos.....	68
6.4.8.	Fontes de Retardos externos .....	68
<b>7.</b>	<b><i>Análise e Implementação das MIBs para gerência ATM .....</i></b>	<b>69</b>
7.1.	<b>MIB II (RFC 1213).....</b>	<b>69</b>
7.2.	<b>AToM MIB (IETF – RFC 1695/2515) .....</b>	<b>72</b>
7.3.	<b>A MIB proprietária IBM ATM-SWITCHING-NODE-MIB v. 4.1.12 .....</b>	<b>74</b>
7.4.	<b>ILMI .....</b>	<b>80</b>
	O protocolo ILMI .....	81
<b>8.</b>	<b><i>O Ambiente de gerência ATRM-WTool.....</i></b>	<b>89</b>
<b>8.1.</b>	<b>Módulo de Inicialização .....</b>	<b>89</b>
8.1.1.	Função .....	89
8.1.2.	Modo de Operação.....	90
8.1.3.	Facilidades apresentadas ao gerente .....	93
<b>8.2.</b>	<b>Módulo de Coletas.....</b>	<b>93</b>
8.2.1.	Função .....	93
8.2.2.	Modo de Operação.....	93
8.2.3.	Facilidades apresentadas ao gerente .....	94
<b>8.3.</b>	<b>Módulo de Alterações/Manutenção (Manager).....</b>	<b>95</b>
8.3.1.	Função .....	95
8.3.2.	Modo de Operação.....	96
8.3.3.	Facilidades apresentadas ao gerente .....	96
<b>8.4.</b>	<b>Módulo de Visualização .....</b>	<b>97</b>
8.4.1.	Função .....	97
8.4.2.	Modo de Operação.....	97
8.4.3.	Facilidades apresentadas ao gerente .....	98
	Banda alocada .....	98
<b>8.5.</b>	<b>Relacionamentos das unidades funcionais com as tabelas do DBMS.....</b>	<b>102</b>
<b>9.</b>	<b><i>Resultados .....</i></b>	<b>104</b>

9.1.	<b>Introdução.....</b>	<b>104</b>
9.2.	<b>Resultados Obtidos.....</b>	<b>105</b>
9.2.2.	Monitoramento das conexões .....	105
9.2.3.	Identificar os Níveis de QoS negociados em cada Conexão.....	107
9.2.4.	Estudar as variáveis presentes nas MIBs definidas pelo ATM Forum e pelo IETF para a gerência de tráfego ATM.....	108
9.2.5.	Identificar os níveis críticos que os parâmetros referentes a cada classe de serviço podem atingir. 108	
9.2.6.	Implementar a ferramenta de gerência ATRM-WTool.....	109
9.3.	<b>Análise Histórica Qualitativa .....</b>	<b>109</b>
9.3.1.	Análise do fluxo de entrada/saída dos VCIs .....	109
9.3.2.	Células com erros .....	111
9.3.3.	Número de Conexões ativas .....	112
9.3.4.	Frequência de saída de células.....	113
9.4.	<b>Dificuldades Encontradas.....</b>	<b>114</b>
9.5.	<b>Conclusões.....</b>	<b>115</b>
9.6.	<b>Sugestões para trabalhos futuros.....</b>	<b>117</b>
10.	<b><i>Bibliografia .....</i></b>	<b><i>120</i></b>
11.	<b><i>Siglas .....</i></b>	<b><i>126</i></b>
	<b><i>Anexo I.....</i></b>	<b><i>129</i></b>
	◆ <b>Grupo atmSvcLogTable .....</b>	<b>130</b>
	◆ <b>Grupo AtmQ2931StatsEntry .....</b>	<b>132</b>
	◆ <b>Grupo nbrTable .....</b>	<b>133</b>
	◆ <b>Grupo vcXConnectTable .....</b>	<b>133</b>
	◆ <b>Grupo interfaceTable .....</b>	<b>136</b>

## Índice das Figuras

Figura 3-1 - Topologia da rede ATM em implantação na UFSC .....	8
Figura 3-2 - Ambiente de estudos.....	10
Figura 4-1 - Conceito do ATM .....	12
Figura 4-2 - Rede ATM Típica .....	14
Figura 4-3 - Representação esquemática dos Canais e Circuitos Virtuais .....	15
Figura 4-4 Comutação de células .....	16
Figura 4-5 - Diagrama lógico da CAC .....	20
Figura 4-6 - Ciclo de vida da SVC e pontos de interferência .....	22
Figura 5-1 - Modelo de relações Gerente-Agentes.....	28
Figura 5-2 - Estrutura em árvore das MIBs .....	30
Figura 5-3 - O papel do SNMP ([55]).....	31
Figura 5-4 - A solução 'Proxy' para a WBM.....	37
Figura 5-5 - Abordagem 'Embutida' para WBM .....	37
Figura 5-6 - Java Virtual Machine .....	39
Figura 6-1 - Módulos de Gerência do ATM Forum .....	44
Figura 6-2 - Conexões virtuais e Links Virtuais .....	48
Figura 6-3 - Relação entre as variáveis de Tráfego .....	49
Figura 6-4 - Gerência da AAL5 em um Comutador ATM.....	50
Figura 6-5 - Gerência da Camada AAL5 em um Host.....	50
Figura 6-6 - Formato do tráfego como policiamento.....	55
Figura 6-7 - Tempo de re-montagem de células em CBR .....	65
Figura 6-8 - Componentes do <i>Cell Transfer Delay</i> .....	67
Figura 7-1 - <i>ifTable</i> parcial para um comutador IBM CPSW 8265 .....	71
Figura 7-2 - Exemplo de amostragem para <i>ifInOctets</i> na interface 101 do comutador CPSW_NPD.....	71
Figura 7-3 - Estrutura da AToM MIB.....	73
Figura 7-4 - Visão parcial da estrutura da MIB <i>IBM ATM-SWITCHING-NODE-MIB</i> v. 4.1.12 .....	76
Figura 7-5 -ILMI versão 4.0 .....	81
Figura 7-6 - Comunicação entre duas IMEs através da ILMI .....	82
Figura 7-7 - Número de VPIs e VCIs.....	84
Figura 7-8 - Definição e contexto da ILMI - UNI 3.1 .....	86
Figura 7-9 - Configuração de proxys SNMP ([55]).....	87
Figura 7-10 - Acesso NMS a ILMI através de agente proxy .....	88
Figura 8-1 - Tela de Abertura do ATRM-WTool.....	89
Figura 8-2 - Relacionamento entre as classes na ferramenta de desenvolvimento Adventnet Builder .....	91
Figura 8-3 - Vetor de variáveis SNMP para objeto <i>SNMPTarget1</i> , elaborado no ambiente de desenvolvimento.....	92
Figura 8-4 - Configuração das variáveis para classe <i>JDBCAdapter</i> .....	94
Figura 8-5 - Tela de comandos SQL do módulo manutenção .....	96
Figura 8-6 - Banda disponível e banda alocada em uma porta com tráfego UBR (bps) .....	99
Figura 8-7 - Banda disponível e banda alocada em uma porta com tráfego UBR mais uma conexão CBR de 50000 Kbps de taxa de pico (PCR).....	100
Figura 8-8 - Banda disponível e banda alocada na mesma porta após o encerramento da conexão CBR.....	100
Figura 8-9 - Fluxo de entrada e saída de células.....	101

<b>Figura 8-10 - Visualização da utilização das CPUs e temperatura ambiente nos comutadores</b> .....	101
<b>Figura 8-11 - Estrutura de armazenamento para parâmetros de tráfego nas portas físicas dos comutadores</b> .....	102
<b>Figura 8-12 - Estrutura da tabela utilizada para monitorar o estabelecimento de conexões comutadas UNI.</b> .....	103
<b>Figura 8-13 - Estrutura da tabela utilizada para monitorar o tráfego nas conexões comutadas, independente da sinalização (UNI, PNNI)</b> .....	103
Figura 9-1 - Fluxo de saída nos VCs .....	110
Figura 9-2 - Fluxo de Entrada nos VCs .....	110
<b>Figura 9-3 - Células com erros</b> .....	111
Figura 9-4 - Número de Conexões ativas .....	112
Figura 9-5 - Erro de amostragem durante a fase de depuração da ferramenta. ....	113
Figura 9-6 - Frequência de saída de células .....	114
<b>Figura 9-7 - Ferramenta Tipo "Sniffer" com funcionalidades para monitoramento de PVCs e SVCs</b> .....	116
Figura 9-8 - Componentes do contrato de tráfego .....	119
<b>Figura I-1 - Direção do fluxo da conexão</b> .....	135

### Índice das Tabelas

<b>Tabela 3-I- Relação dos equipamentos da rede ATM – UFSC (Agosto de 1999)</b> .....	9
<b>Tabela 5-1 - Diferenças de nomenclatura na Gerência de Tráfego</b> .....	26
Tabela 6-1 - MIBS do Modelo ATM Forum [35] .....	47
Tabela 6-2 – Categorias de serviço da camada ATM.....	56
Tabela 6-3 - Valores do fator <i>a</i> em diversas redes .....	64
<b>Tabela 7-1 - Tipos de interfaces significativas para o ATM</b> .....	70
<b>Tabela 7-2 - Formato DateTime (RFC 1443)</b> .....	77
<b>Tabela 7-3 - Valores retornados pela variável <i>interfaceFrameFormat</i></b> .....	79
Tabela 7-4 - Módulos da MIB ILM1 .....	83
<b>Tabela 8-1 - Descrição das variáveis SNMP, MIBs e posição no vetor de inicialização.</b> .....	92
<b>Tabela 8-2 - Módulos Funcionais do ATM</b> .....	99
<b>Tabela 9-1 – Lista atual de Drafts do AToM Working Group do IETF [45].</b> .....	118
<b>Tabela I-1 - Valores retornados pelas variáveis <i>AtmSvcLogForwardQOS</i> e "<i>AtmSvcLogBackwardQOS</i></b> .....	131
<b>Tabela I-2 - Valores de retorno para a Categoria de serviço</b> .....	131
<b>Tabela I-3 - Valores de retorno para o local de solicitação de encerramento da conexão</b> .....	132
<b>Tabela I-4 - Exemplo de Conexão Cruzada</b> .....	135
<b>Tabela I-5- Valores retornados pela variável <i>interfaceOperState</i></b> .....	139
<b>Tabela I-6 - Valores de retorno para Tipo de acesso ATM</b> .....	140

---

## 1. Introdução

---

Acompanhando o desenvolvimento dos sistemas computacionais, pode-se constatar que as redes tornaram-se um segmento essencial da ciência da computação. As aplicações de rede, como o correio eletrônico, a transferência de arquivos, a visualização das informações usando a WWW, os sistemas de 'Workgroup Computing', 'Workflow', processamento distribuído, videoconferência, telemedicina, tomam um tempo bastante elevado dos recursos de um computador, quando comparados com as demais atividades computacionais [34] [36].

As redes de computadores atuais, utilizando as tecnologias do tipo Ethernet, FDDI e Token Ring (na camada de enlace do modelo de referência OSI - [58]) e o TCP/IP (para a camada de rede e transporte do modelo OSI), não atingem plenamente os objetivos das aplicações citadas anteriormente. Isso ocorre especialmente nas aplicações que envolvem diferentes tipos de tráfego (voz, vídeo e dados) [6]. O foco dos estudos dos meios de transmissão de sinais tem evoluído na direção de uma tecnologia única, capaz de atender as demandas por desempenho e transmissão de tipos variados de tráfego. Várias tecnologias têm sido desenvolvidas para resolver o problema [16] [50], uma vez que diferentes tipos de tráfego exigem redes com características diferenciadas.

O ATM (*Asynchronous Transfer Mode*), tem sido a solução de rede escolhida para uma parcela considerável das instituições principalmente aquelas que estiverem buscando:

- Aumentar o desempenho almejado pela computação científica
- Garantir a qualidade de serviço (QoS);
- Propiciar tráfego multimídia (voz, dados e vídeo);
- Possibilitar a utilização dos conceitos de rede virtuais (VLANs);
- Otimizar as funções de roteamento.

A tecnologia ATM deve dominar o tráfego de vídeo e áudio em redes WAN e, devido a sua grande flexibilidade, tende a ser a tecnologia dominante para esses tráfegos também nas redes locais [6].

A definição dos padrões para essa tecnologia está centralizada em dois organismos de padronização:

- **ITU-T**, ramo de padronização de telecomunicações da *International Telecommunications Union (ITU)*, que está creditando à tecnologia ATM a responsabilidade de transportar os serviços da B-ISDN (*Broadband Integrated Digital Network* - Rede Digital com Integração de Serviços – Banda Larga), principalmente no que se refere à parte pública das redes das provedoras de serviço.
- **ATM Forum**, um consórcio formado por várias empresas e usuários, cujos maiores objetivos são desenvolver a tecnologia e padronizar as redes ATM privadas.

O ATM Forum procura alinhar seus padrões com aqueles definidos pelo ITU-T para as redes de telecomunicações ATM. Apesar dos esforços de padronização, soluções proprietárias surgiram antecipadamente, o que torna difícil a interoperabilidade e gerência dos equipamentos de diferentes fabricantes [19] [6].

### ***1.1. Descrição do problema***

A monitoração de recursos e tráfego em redes ATM é uma tarefa complexa [4] [15] [43] [47], devido ao grande número de serviços, características de tráfego, escalas de tempo, e restrições de performance., as quais estão integradas nos sistemas da rede [20] de forma que os avanços na tecnologia resultam em rápidas mudanças nos maiores “gargalos” das redes.

As aplicações de gerência tradicionalmente são restritas a uma estação dedicada e acessada somente através de equipamentos com alta capacidade de processamento para a visualização da interface gráfica. Exigiam, além disso, treinamento e conhecimentos especializados.

A grande maioria das ferramentas atuais de gerência foi especificada para redes não ATM [2]. Além disso os padrões de gerência para ATM são recentes e continuam sem ser implementados pelos fabricantes. Essa fase de desenvolvimento das tecnologias leva os fabricantes a desenvolverem extensões proprietárias para seus agentes gerenciáveis. Isso leva a uma situação onde a tentativa de gerenciar equipamentos de diferentes fabricantes pode ser bastante difícil.

Os problemas da gerência ATM começam nas taxas de transmissão, mais elevadas e diversificadas. Isso significa que quantidades bem maiores de informações serão emitidas para a estação de gerência em um dado intervalo de tempo, possivelmente em grandes rajadas. Deve-se considerar também que os diferentes tipos de tráfego envolvidos pela tecnologia acrescentam mais informações a serem

monitoradas. Pela abordagem tradicional, o armazenamento de grandes volumes de dados é um fator limitante. Esse problema precisa ser atacado para que se possa montar uma *baseline*<sup>1</sup> do comportamento da nossa rede, a partir da qual os parâmetros monitorados possam ser correlacionados. Um estudo dos padrões de comportamento dessa massa de dados também se faz necessária para a tomada de decisões. Iniciou-se este estudo com uma amostragem a cada 300 segundos, durante um período de 15 dias. Um tratamento estatístico descritivo básico está demonstrado na seção 9.

Talvez a principal diferença entre a gerência de redes tradicionais resida no fato de que, ao gerenciar uma rede ATM, o gerente deve ter bem claro que tratam-se de duas redes distintas: uma física e outra virtual. Essa última tem suas complexidades inerentes, seja pela quantidade de conexões ou pela informação de sinalização que transita pela rede nos diferentes planos do modelo de referência.

As estações de gerência (NMS) também devem ser capazes de suportar grandes volumes de dados. E as aplicações de gerência devem ter capacidade de correlacionar as informações, tornando fácil a detecção dos problemas críticos, sem necessidade de “garimpar” esses dados. A aplicação de gerência deve estar adaptada a topologia da rede, podendo dessa forma classificar os dados pela localização dos comutadores. Maiores detalhes sobre a gerência ATM estão descritos na seção 5.5.

## ***1.2. Descrição da proposta***

O presente estudo faz uma análise dos parâmetros de tráfego necessários ao gerenciamento de recursos das redes ATM. Essa avaliação envolve as bases de dados de gerenciamento (MIBs) para monitoração e controle de recursos como largura de banda e disponibilidade de canais virtuais para novas conexões, o controle do estabelecimento dessas conexões, suas origens e quais as estações conectadas.

Com base nos parâmetros estudados, foi implementado um ambiente que possibilita a gerência de tráfego e recursos da rede através de uma interface única, independente do fabricante do equipamento e da plataforma utilizada. Baseado na WWW (World Wide Web), na linguagem Java e no protocolo SNMP (Simple Network Management Protocol), o *ATRM-WTool* (ATM Traffic and Resources Management Web Tool) se propõe a fornecer aos gerentes de rede ATM a facilidade de controle e

---

<sup>1</sup> O termo indica um levantamento histórico do comportamento da rede, a partir do qual se pode assumir determinados compromissos entre os limites normais e anormais.

---

monitoração de alguns parâmetros relativos ao tráfego e recursos da rede a partir de qualquer estação conectada à Internet.

O ambiente de gerência propicia ainda uma análise histórica e estatística dos dados, pois armazena o resultado das coletas em uma base de dados relacional, permitindo acesso a esses dados, tanto aos gerentes (de uma forma mais ampla, através de consultas livres) como aos usuários finais da rede, aos quais é disponibilizada uma verificação de tráfego. Dessa forma, o comportamento da rede ao longo do tempo pode ser analisado em detalhes.

A abordagem do trabalho possui ainda vantagens sobre ferramentas de gerência atuais: Integrando agora a parte cliente a um browser WEB, a aplicação de gerência, instalada na estação servidora, passa a ser acessível a todos os usuários, em todos os pontos da rede, com mínima necessidade de treinamento. Tomando-se alguns cuidados básicos em relação à segurança, todos os usuários podem ter acesso à informações de tráfego e recursos nos circuitos da rede ATM.

Pode-se verificar, em uma rápida análise do mercado atual, uma tendência dos fabricantes dessas plataformas clássicas em migrar para a tecnologia Web (IBM-Tivoli, 3Com-Transcend, Cabletron Spectrum-Web). Na literatura, várias soluções de gerência de rede integrada na Web podem ser encontradas [6] [9] [17] [28] [42] [67].



---

## 2. Objetivos

---

### 2.1. *Objetivos Gerais*

Um dos objetivos gerais deste estudo foi a implantação do Backbone ATM na rede de computadores da Universidade Federal de Santa Catarina. Tal rede será denominada simplesmente de redeUFSC no escopo deste estudo. Após a fase de implantação da rede ATM, objetivou-se a implantação de um ambiente de gerência para as redes ATM, procurando *monitorar e controlar* o tráfego e recursos disponíveis para as conexões ATM, principalmente as conexões comutadas. Esse ambiente utiliza os protocolos SNMP (*Simple Network Management Protocol*) e HTTP (*Hyper Text Transmission Protocol*). A ferramenta foi desenvolvida na linguagem Java, e utiliza uma arquitetura cliente-servidor. As estações clientes solicitam acesso aos links da rede para a estação servidora, através do protocolo HTTP. Applets Java sendo executados no Browser do cliente (Netscape, Internet Explorer) são os responsáveis pela interface da ferramenta. Recebendo uma invocação do usuário, a aplicação gerente emite e recebe, via SNMP, as primitivas de gerência aos agentes, instalados nos equipamentos da rede. (Sobre essa perspectiva, os agentes assumem o papel de servidores na relação cliente-servidor clássica). As informações da rede são armazenadas em um gerenciador de banco base de dados, para levantamentos estatísticos e históricos.

Esse ambiente de gerência é composto pelos seguintes itens:

- Uma ferramenta de gerência composta por agentes e gerentes (*ATRM-WTool*)
- Biblioteca de gerência SNMP para HTTP (Advent SNMP Package).
- Browsers WEB;
- Banco de dados relacional com linguagem de acesso padrão SQL (*Structured Query Language*).

### 2.2. *Objetivos Específicos*

#### 2.2.1. **Implantação da rede**

Durante a fase de implantação foram efetuados levantamentos de interação dos equipamentos, opções de roteamento e estudo da topologia da rede, com as segmentações necessárias ao melhor aproveitamento dos recursos das redes ATM da

---

Universidade Federal de Santa Catarina (redeUFSC), Rede Metropolitana de Alta Velocidade de Florianópolis (REMAV) e Rede Catarinense de Ciência e Tecnologia (RCT-SC)

### **2.2.2. Monitoração do estabelecimento das conexões virtuais (Rotas Virtuais-VPs e Canais Virtuais-VCs).**

### **2.2.3. Identificação dos Níveis de QoS negociados em cada Conexão.**

### **2.2.4. Estudo das variáveis das MIBs**

Pretende-se identificar os objetos presentes nas MIBs (Management Information Base) definidas pelo ATM Forum e pelo IETF (Internet Engineering Task Force) para a gerência de tráfego e recursos dos comutadores ATM.

### **2.2.5. Identificação dos níveis críticos dos parâmetros**

Os níveis de utilização dos recursos das redes serão monitorados, de forma que alguns parâmetros mais críticos sejam percebidos sempre que se aproximarem dos limites desses recursos.

### **2.2.6. Especificar e implementar o ambiente de gerência *ATRM-WTool* (ATM Traffic and Resources Management Web Tool)**

O ambiente foi desenvolvido com a intenção de monitorar os eventos descritos por Abusamra (1998) [2], segundo o qual os pontos chaves em uma rede ATM são os seguintes:

- Eventos no nível dos Links:
  - Utilização do link
  - Erros de paridade no nível SONET
  - Perda de frames no nível SONET
  - Perda de Sinal no nível SONET
- Eventos no nível de portas:
  - Número de células transmitidas
  - Número de células recebidas
  - Banda alocada
  - Banda utilizada

---

## Retardo das células<sup>2</sup>

- Eventos no nível de Comutador:
  - Status do comutador
  - Status dos módulos do comutador
  - Status da comunicação Host-to-Link
  - Status do link Comutador para Comutador
  - Status das fontes de energia do Comutador
  - Temperatura do Comutador
  
- Eventos no nível de conexões virtuais
  - Número de células transmitidas
  - Banda alocada
  - Banda utilizada
  - Células rejeitadas
  - Retardo das células

---

<sup>2</sup> Embora citado como um dos eventos chave pelo autor, e em diversas referências bibliográficas, não foi possível determinar parâmetros relativos aos diversos retardos encontrados pelas células.

### 3. O ambiente de estudos

#### 3.1. A estrutura ATM na UFSC

A necessidade de desenvolver uma ferramenta visando as metas acima vem da proposta da Universidade Federal de Santa Catarina (UFSC) em migrar seu *backbone* para a tecnologia ATM. A UFSC está necessitando dos novos serviços disponibilizados pelas redes de alto desempenho para, por exemplo, operacionalizar um *Cluster* de estações para o ambiente de computação científica e paralela e obter a largura de banda necessária para as aplicações de telemedicina, videoconferência e vídeo sob demanda.

A rede ATM na UFSC está em fase de implantação e é composta pelos equipamentos (comutadores de borda e núcleo) relacionados na Tabela 3-1. Na Figura 3-1 estão representadas as conexões ATM entre os diversos centros de pesquisa da UFSC. Além desses equipamentos, estão em fase de interoperabilidade os comutadores centrais da redeUFSC com a rede Metropolitana de Alta Velocidade de Florianópolis (RMAV-FLN) e a Rede Catarinense de Ciência e Tecnologia (RCT-SC). Nessa rede de alta performance pretende-se implantar o ambiente de gerência de tráfego descrito nesse trabalho.

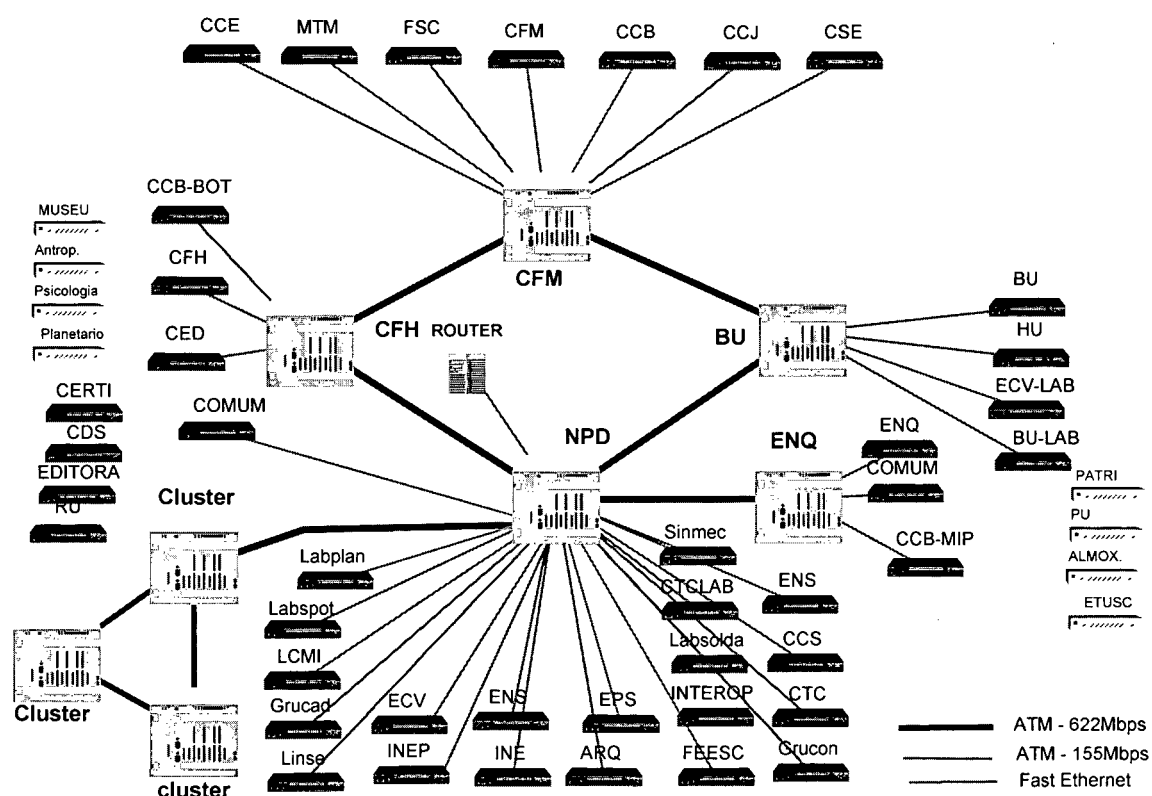


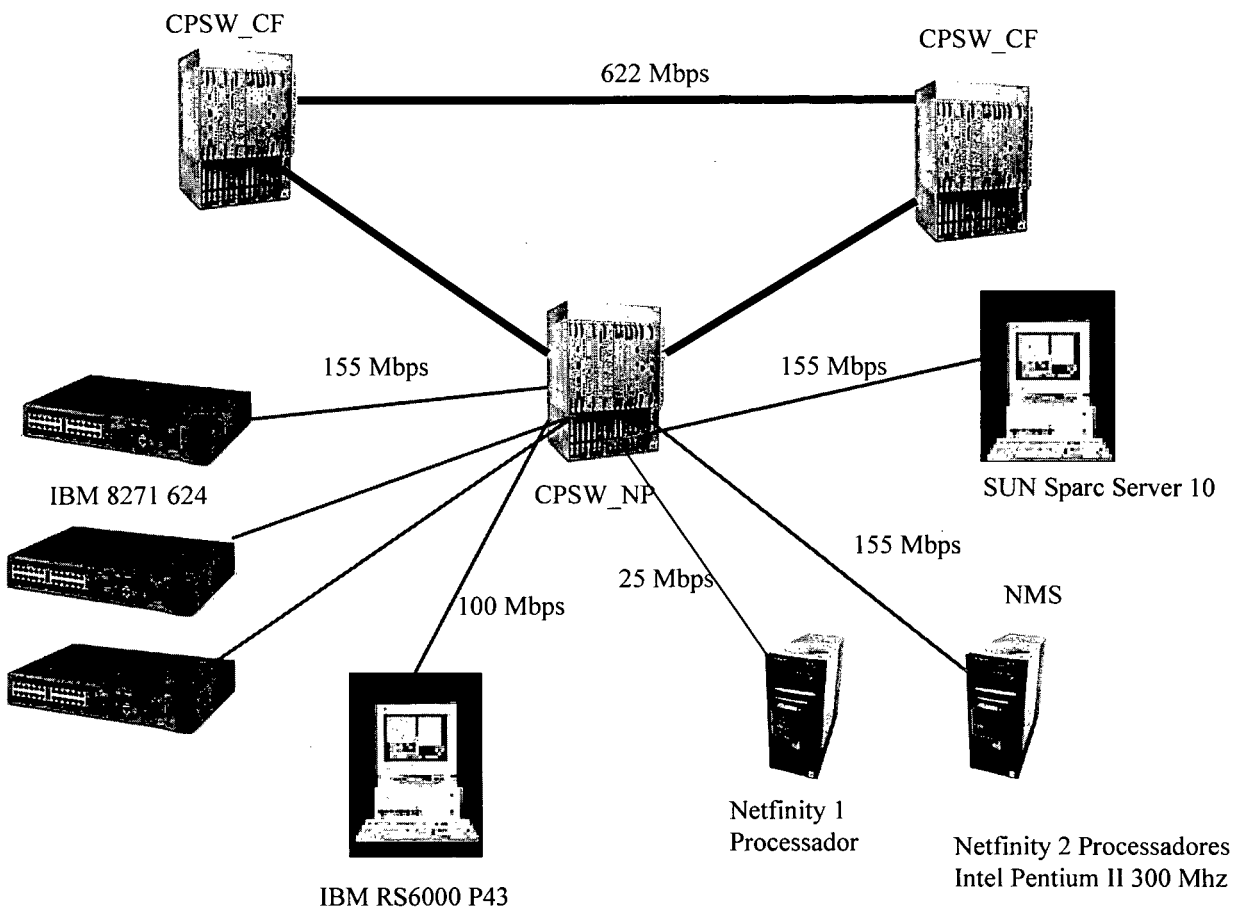
Figura 3-1 - Topologia da rede ATM em implantação na UFSC

Equipamento	Quant.	Portas Eth	Portas 25Mbps	Portas 100MBps	Portas OC3	Portas OC12
IBM 8260	1			8	12	
IBM 8265	3			8	72	8
IBM 8271	20	240			20	
3Com LS 3000	5	60			5	
3Com CB 7000 HD	2	2			40	
Cabletron SS2200	2	48			2	

**Tabela 3-1– Relação dos equipamentos da rede ATM – UFSC (Agosto de 1999)**

### **3.2. Descrição do Ambiente de Estudos**

O estudo descrito neste trabalho foi realizado em um segmento do backbone da rede ATM da Universidade Federal de Santa Catarina (**Figura 3-1**). Para restringir o universo de amostragens, se optou pela utilização de apenas três Comutadores centrais, denominados CPSW\_NPD, CPSW\_CFM e CPSW\_CFH, fabricados pela IBM, modelo CPSW 8265. Além desses Comutadores centrais foram incluídos no ambiente 3 comutadores de borda modelo 8271, todos eles do mesmo fabricante. Como *sistema final ATM*, foram incluídas 4 estações de trabalho: Uma estação *RISC PPC RS 6000 IBM*, dotada de interface (NIC) *ATM IBM TurboWays 100Mbps Fiber Adapter*. Uma estação *Sun Sparc Station 10* equipada com *NIC SunATM SBus Adapter*. Um PC modelo IBM Netfinity, com dois processadores *Intel Pentium II 300, 256 Mb RAM*, equipado com *NIC 3Com ATMLink 155 PCI NIC*. Este PC desempenhou também o papel de servidor Web e propiciou um ambiente de desenvolvimento bastante satisfatório, apesar das grandes exigências de performance dos pacotes de software. Outro PC Netfinity com uma interface *IBM TurboWays 25 PCI NIC* foi utilizado para o estabelecimento das conexões permanentes. Esse ambiente está demonstrado na **Figura 3-2**.



**Figura 3-2 - Ambiente de estudos**

---

## 4. Introdução à Tecnologia ATM

---

Para entender-se os processos envolvidos na gerência de redes ATM, bem como da fusão dessa gerência com a tecnologia Web, é necessária a revisão de alguns conceitos fundamentais das tecnologias envolvidas. Neste capítulo são demonstrados os conceitos básicos do ATM. No capítulo 5, revisaram-se aspectos gerais de gerência de redes. A gerência de redes ATM, especificamente, é discutida no capítulo 5.5, que também apresenta uma revisão da tendência das aplicações de gerência em utilizar a tecnologia Web em conjunto com o protocolo SNMP.

### 4.1. *O que é o ATM*

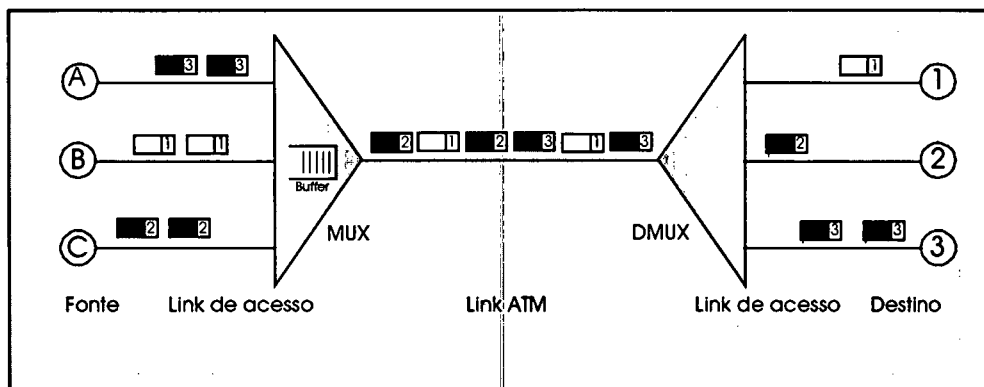
O ATM (*Asynchronous Transfer Mode* ou Modo de Transmissão Assíncrono) é uma tecnologia de comunicação que utiliza um protocolo de comutação de circuitos de alta velocidade. O ATM fornece a capacidade de transmissão de sinais em velocidades que podem chegar a 2.048 Mbps ou maiores [36] com um retardo mínimo, e apresenta a garantia da *Qualidade de Serviço (QoS)*<sup>3</sup>, descrita com algum detalhe na Seção 6.2, página 48. Essa qualidade de serviço estabelece um padrão confiável às transmissões, pois os protocolos irão respeitar os parâmetros estabelecidos para cada conexão. A existência de diferentes qualidades de serviço permite diferenciar os tipos de tráfego, propiciando uma utilização eficiente dos recursos da rede [57] [20] [32]. Do ponto de vista do gerente de tráfego essa divisão lógica do fluxo é a única maneira de controlar, em um mesmo meio físico, tipos de tráfego tão variados quanto os permitidos pelo ATM [20].

O ATM transfere pacotes de tamanho fixo de 53 bytes, denominados células. Esse tamanho fixo permite que toda comutação seja feita por hardware, evitando procedimentos de software, menos eficientes [38]. O tamanho fixo simplifica a arquitetura do comutador, uma vez que a largura de banda solicitada por uma conexão não influencia no algoritmo de comutação implementado no equipamento [20]. A

---

<sup>3</sup> *Quality of Service* - Parâmetros de serviço para uma conexão ATM, especificando características como taxa de perda de células e retardo de transferência de células.

transmissão é feita em modo orientado à conexão<sup>4</sup>. O fluxo de células de uma conexão ATM é assíncrono. Isso permite que o fluxo de dados seja inserido no meio físico em qualquer instante, sem necessidade de uma estação esperar seu intervalo de tempo para transmitir. Na transmissão assíncrona, a identificação do canal e das estações de origem e destino é feita por um cabeçalho inserido nos pacotes. Diferentes fluxos de células, de conexões com requerimentos diversos de QoS podem ser multiplexados em um mesmo link físico.



**Figura 4-1 - Conceito do ATM**

Atualmente, novas especificações do ATM Forum permitem “Multiplexação Inversa”, onde mais de um link físico (por exemplo, linhas T1/E1, ou outros links de baixo custo) são agregados em circuitos virtuais únicos [70], permitindo aumentar a largura de banda. O ATM trabalha com a combinação de duas funções simples:

- Comutação de células baseada em Hardware.
- Estabelecimento de conexão baseada em Software.

#### 4.2. *Objetivos do ATM*

O Modo de Transmissão Assíncrono é uma tecnologia baseada na comutação de células e multiplexação, projetada para ser orientada a conexão e abranger uma ampla gama de serviços de propósitos gerais. O ATM está sendo aplicado também às tecnologias de redes locais e privadas, como especificado pelo ATM Forum.

As conexões virtuais do ATM podem operar com várias categorias de serviços, como *Constant Bit Rate (CBR)*, *Variable Bit Rate (VBR)* e *Unspecified Bit Rate (UBR)*,

<sup>4</sup> Comunicações orientadas à conexão exigem que o endereço de destino aceite a recepção do fluxo antes que ele inicie, e isso permite o estabelecimento de rotas antecipadamente, inclusive com QoS negociada, se as características da rede permitirem.



ou *Available Bit Rate (ABR)* [47] [31] [57] [20]. O controle de tais conexões é um dos focos de estudo do presente trabalho (Seção 6.2). Cada célula ATM transmitida na rede possui informações de endereçamento que estabelecem uma conexão virtual da origem para o destino. As células são transmitidas seqüencialmente através dessa conexão virtual, que pode ser estabelecida manualmente pela configuração dos comutadores (Canal Virtual Permanente - PVC ou *Permanent Virtual Circuit*) ou Comutada (SVC – *Switched Virtual Circuit*). As conexões comutadas são estabelecidas através dos recursos de sinalização da rede ATM, enquanto as permanentes podem ser estabelecidas somente pelo gerente da rede, seja através de uma aplicação de gerência ou através da linha de comandos dos comutadores ATM.

O ATM oferece potencial para padronizar, em uma só arquitetura, as definições de métodos de multiplexação e comutação, bem como integrar as tecnologias aplicadas em LANs, MANs e WANs [34].

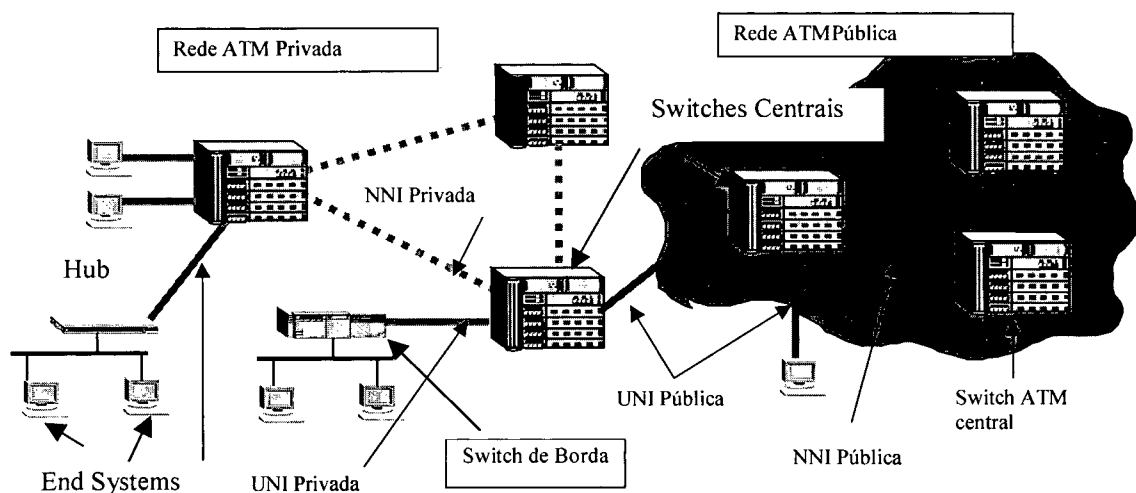
Como o ATM suporta diferentes classes de Qualidade de Serviço (QoS), a alocação de largura de banda pode ser alterada dinamicamente, conforme a demanda, através dos Circuitos Virtuais Comutados (SVCs). Assim uma rede inteira pode ser construída usando-se a tecnologia ATM para suportar a comutação e multiplexação de serviços como:

- Voz
- Pacotes de Dados (IP, Frame-Relay)
- Vídeo
- Imagens
- Emulação de Circuitos (circuitos virtuais)

### **4.3. Topologia das redes ATM**

Uma rede ATM consiste de um conjunto de dispositivos ATM interconectados (Figura 4-2). Os hosts ATM são Workstations ou PCs equipados com uma Interface de Rede (NIC – *Network Interface Card*) ATM. Diferente das tecnologias de rede que compartilham o meio físico, como a Ethernet (através do método de acesso CSMA/CD<sup>5</sup>), o ATM requer um comutador para interconectar os sistemas finais (*sistema final ATM*). Um comutador apresenta uma matriz de portas para conectar os

sistema final ATM ou outros comutadores. Existem comutadores de pequena escala (CS, *Campus Switch*) para as redes privadas (com 10 a 256 portas) e comutadores de larga escala (CO, *Central Office*), usados pelas companhias de telecomunicações para comportar as redes ATM públicas (até 1000 portas, como o modelo *Corex*, fabricado pela Motorola) [38]. Portas diferentes podem suportar meios de transmissão diferentes, como par trançado não blindado (UTP) ou coaxial de cobre, mas usualmente o meio físico utilizado é a fibra óptica, pois permite taxas de transmissão mais elevadas. Redes ATM locais (ATM LANs) podem ser conectadas às LANs tradicionais via LAN-Comutadores ATM, que fazem a tradução dos protocolos na camada de enlace de dados.



**Figura 4-2 - Rede ATM Típica**

Dois tipos de interface foram definidos para interconectar dispositivos ATM adjacentes [62]:

- User Network Interface (UNI), que suporta conexão entre sistemas finais ATM (hosts, roteadores) e os comutadores.
- Network - to - Network Interface (NNI), que define a conexão entre dois comutadores.

Dependendo do operador (público ou privado) entre os comutadores adjacentes, pode-se distinguir as interfaces NNI e UNI como públicas ou privadas. As interfaces

<sup>5</sup> Common Shared Media Access / Collision Detect – Algoritmo básico para acesso em redes padrões Ethernet, Fast-Ethernet e Gigabit Ethernet, os quais são baseados em compartilhamento do meio físico e detecção de colisões.

privadas têm sido padronizadas pelo ATM Fórum, enquanto as públicas são responsabilidade do ITU-T.

Uma rede ATM tem uma topologia de malha em estrela [38]. A interconexão dos comutadores que formam o núcleo (*core*) da rede é feita em malha. As bordas, nas UNI em volta desse núcleo, formam a estrela.

#### 4.4. *VPI e VCI*

Cada célula contém os números Virtual Path Identifier (VPI) e Virtual Channel Identifier (VCI), que especificam a conexão virtual por onde ela deve ser transportada. Um canal virtual (VC) é um circuito (Figura 4-3) que transporta células entre dois ou mais nós. Quando vários VCs usam o mesmo caminho de transmissão, eles podem ser agrupados num único Virtual Path (VP).

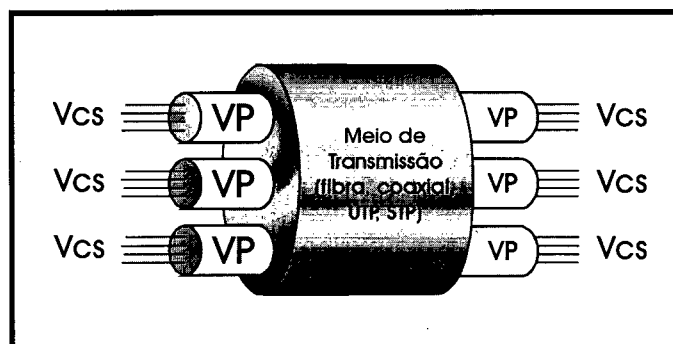


Figura 4-3 – Representação esquemática dos Canais e Circuitos Virtuais

#### 4.5. *Comutadores (switches)*

Quando uma célula chega no comutador, os números VCI/VPI da célula e a porta por onde ela entrou são comparados com valores de uma tabela (Figura 4-4) para determinar por qual porta de saída a célula deve ser enviada.

Os números VPI/VCI têm significado local apenas, uma vez que os valores podem ser modificados em cada um dos comutadores pelos quais a célula é transportada.

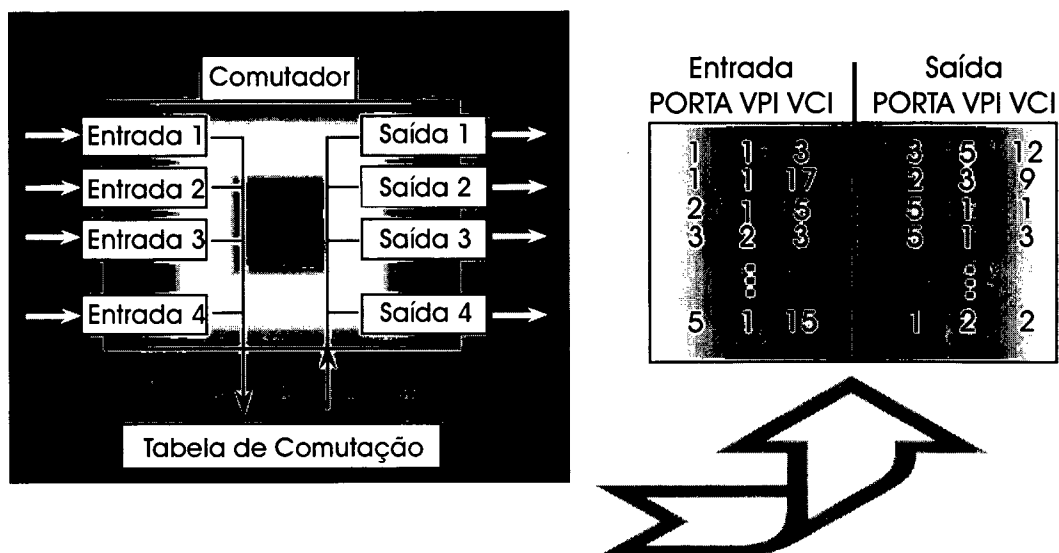


Figura 4-4 Comutação de células

#### 4.6. Sinalização nas redes ATM

Sinalizar é o ato de comunicar-se por sinais em tempo real. Um sinal é um símbolo ou meio de comunicação acordado ou entendido pelas entidades envolvidas de forma a trocar informações ou comandos entre as entidades comunicantes.[44]. Em redes de comunicação a sinalização permite aos usuários da rede comunicarem seus requerimentos de serviços para a rede (sinalização usuário-rede). Da mesma forma, ela permite que os dispositivos de rede (comutadores, servidores) troquem entre si todos os tipos de informações necessários para manter a rede e dar suporte ao tráfego dos usuários (sinalização rede-rede).

Nem todas as redes de comunicação utilizam sinalização. Por exemplo, a sinalização não pode ser utilizada para emitir um pacote de uma estação para a outra através de uma rede local tradicional (*Legacy LAN*), ou um pacote através de um serviço de datagramas. Nestas redes, em termos simplificados, uma fonte de uma estação terminal envia seus pacotes para o dispositivo ao qual está conectada. Os pacotes são roteados de um dispositivo da rede para outro até que atinjam seu destino. Cada dispositivo de rede observa o endereço de destino incluso no pacote e usa essa informação para, em conjunto com as informações da rede que ele “conhece”, decidir o próximo destino do pacote. Como a visão que um dispositivo possui sobre sua rede pode mudar no decorrer do tempo, dois pacotes que possuem os mesmos endereços de fonte e destino podem seguir diferentes caminhos através da rede.

Entretanto, nem todas as redes de comutação de pacotes oferecem serviços de melhor esforço. Redes comutadas por pacotes que possuem a tecnologia de circuitos virtuais precisam estabelecer uma rota fim-a-fim antes de iniciarem o fluxos de transmissão das informações. (Mas tal técnica não exige, necessariamente, recursos dedicados de comunicações para inicializar). Neste caso, os identificadores de conexão são configurados e reservados no acesso a rede e/ou em toda a rede para servirem de identificadores únicos da conexão. Tal variável pode ser coletada através da MIB de gerência proprietária da IBM, através do objeto *AtmSvcCallReference*, que retorna o valor de referência da chamada pela especificação Q2931 do ITU-T [33], utilizada por esta Switched Virtual Connection.

As s tecnologias X.25, Frame-Relay, e Systems Network Architecture (SNA) são exemplos de tecnologias orientadas para circuitos virtuais.

A UNI (User Network Interface) é o ponto de demarcação entre uma sistema final (end-system) e a rede. Neste contexto um sistema final ATM pode ser um computador pessoal, um comutador, um gateway ou outro equipamento capaz de rodar protocolos ATM. A especificação ITU-T Recommendation Q.2931 [33] contém os procedimentos utilizados para o estabelecimento, manutenção e desligamento das conexões no nível da UNI . As especificações do ATM Forum para a UNI, versões 3.1 e 4.0 [62] são baseadas na especificação Q.2931, com várias extensões para suportar características que podem ser utilizadas em redes ATM privadas. Cada conexão estabelecida na camada ATM é bi-direcional, e as características da conexão cruzada podem ser diferentes em cada direção da conexão. Pode-se citar como exemplo uma conexão entre uma televisão e um servidor de vídeo. Apenas alguns Kbps de largura de banda são necessários da televisão para o servidor de vídeo, enquanto vários megabits podem ser necessários na direção contrária.

Aplicações diferentes possuem requerimentos diferenciados de serviços. Pela especificação Q.2931, é permitido que cada direção da conexão possua diferentes classes de QoS em cada direção. Um requerimento de QoS para a conexão é especificado usando uma classe individual ou parâmetros de QoS apropriados para a aplicação de rede. Enquanto a versão 3.1 da UNI suporta somente as classes de QoS, a versão 4.0 suporta também parâmetros diferenciados.

Os procedimentos de sinalização são utilizados pára gerenciar canais virtuais sob demanda, em tempo real. Uma vez estabelecida, uma conexão comutada pode

---

permanecer ativa por um período de tempo arbitrário. A estação final que originou a conexão é denominada “origem”, (*calling party*), e a estação de destino, *called party*.

#### **4.7. Controle de Admissão de Conexão – CAC**

O ganho na multiplexação estatística do sinal (**Figura 4-1** - [20]), que pode ser obtido pela superposição de várias fontes em um mesmo canal de transmissão é uma das vantagens mais salientes do ATM sobre tecnologias síncronas ou de comutação de circuitos.

O aumento indiscriminado do número de fontes que compartilham o mesmo meio de transmissão pode degradar os serviços prestados pela rede, causando congestionamento e perda de células. Alguns mecanismos de controle de tráfego são necessários para evitar esses problemas e garantir a qualidade do serviço (QoS) contratada. Os parâmetros de QoS como a taxa média de transmissão (MCR), taxa de pico (PCR) devem ser especificados e monitorados pela rede.

O conceito de controle de tráfego nas redes ATM passa através de algumas fases bem definidas: O usuário (aplicação) declara suas necessidades no momento do estabelecimento da conexão. A rede, para aceitar essa conexão, deve assegurar as características negociadas. Caso isso seja impossível, a solicitação de conexão é rejeitada. Durante a conexão, a rede monitora a conformidade entre o que foi negociado e aquilo que a aplicação está utilizando. Se não houver conformidade, algum tipo de penalidade deve ser atribuída a aplicação.

O mecanismo de rede que controla as conexões é denominado CAC (*Connection Admission Control*) e o mecanismo que controla o fluxo de células é o UPC (*Usage Parameter Control*).

##### **4.7.1. Serviço de Switched Virtual Connection (SVC, ou Conexão Virtual Comutada)**

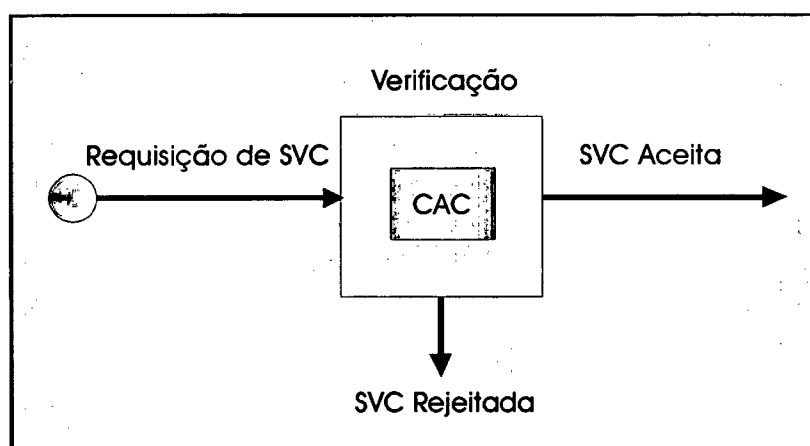
A utilização de conexões comutadas, que são solicitadas pela aplicação do usuário, podem consumir recursos importantes da rede, que são compartilhados e finitos. É necessário manter um policiamento sobre esses recursos, para poder-se balancear as cargas na rede, bem como estimar-se possíveis necessidades de atualização ou detectar-se eventuais sobrecargas ou “gargalos” na rede [24] [53] [61]. Algumas considerações devem ser observadas a respeito das SVCs, devido sua importância para o policiamento do tráfego.

As conexões Comutadas possibilitam ao usuário estabelecer circuitos virtuais sob demanda. Existem diferenças relevantes entre os termos usuário e estação terminal (*end-station* ou *end-system*). O usuário deve ser considerado como o ser humano. O termo *end-system* especifica um sistema computacional que está no final de um *link* ATM. Um *sistema final ATM* pode ser referenciado pelo seu endereço ATM. Portanto, uma inicialização de SVC sempre será efetuada pelo usuário, e a conexão comutada resultante sempre irá conectar dois ou mais *sistema final ATM*.

Uma SVC sempre requer recursos da rede. Esses recursos incluem:

- Largura de Banda nos links da rede;
- Valores disponíveis de VPI e VCI;
- Espaço de buffer para as células nos comutadores ATM.

Quando um usuário requisita uma nova SVC, a rede irá verificar a disponibilidade de recursos para a conexão. Se existirem, a conexão é aceita. Caso contrário, a conexão é rejeitada. Essa função de controle é a *Call Admission Control* (CAC, ou Controle de Admissão de Conexões), que pode ser representada pelo diagrama simplificado na **Figura 4-5**.



**Figura 4-5 - Diagrama lógico da CAC**

#### 4.7.2. A necessidade de uma política para as Conexões Comutadas

O serviço de SVC disponível atualmente nos comutadores ATM invoca a *Call Admission Control*. Dessa forma, uma requisição de SVC somente será aceita se a rede possuir recursos suficientes para garantir a QoS solicitada pela SVC. A existência somente dessa função de controle para as SVC não é suficiente para uma rede ATM em atividade de produção, como é o caso do ambiente em estudo [53]. Mecanismos



adicionais são necessários para controlar a aceitação das SVCs e, dessa forma, controlar a maneira como os recursos finitos da rede serão divididos entre os usuários. Pode-se chamar esses mecanismos adicionais de “política de aceitação de SVC” ou simplesmente “política de SVC”.

Uma rede ATM que controle suas conexões comutadas unicamente através da CAC pode ter sua política de SVC descrita como “o primeiro que pede, consegue”. Todos os usuários possuem chances iguais na obtenção dos recursos da rede. A qualquer momento, um usuário pode requisitar grandes porções de recursos e, uma vez tendo conseguido, pode permanecer com eles por períodos de tempo indeterminados e arbitrários. Esse tipo de situação pode ser considerada inaceitável em alguns tipos de redes. Por exemplo, alguém poderia solicitar uma conexão de 100 Mbps domingo a noite, quando houvessem recursos. A CAC iria admitir a conexão, e na segunda-feira os recursos estariam “esgotados” para novas conexões. Portanto, são necessários outros mecanismos mais refinados que o simples “o primeiro que pede, consegue” implementado pela CAC.

#### **4.7.3. Comparação entre circuitos Permanentes e Comutados (PVC/SVC)**

Uma aplicação de gerência também pode ser usada para estabelecer conexões virtuais nas redes ATM, ao invés de sinalização. Tais conexões são denominadas PVCs, ou *Permanent Virtual Connections*. Essas conexões também necessitam recursos da rede, e a princípio, podem ser tratadas pela mesma política de controle que suas parceiras comutadas. A principal diferença entre as PVCs e SVCs é que atualmente, as PVCs não podem ser estabelecidas pelo usuário final. Elas somente são estabelecidas através de uma ação de gerenciamento sobre a rede. Dessa forma, a decisão de quando aceitar ou estabelecer a conexão permanente passa pelo gerente da rede. O gerente deve verificar se os recursos necessários estão disponíveis e se a conexão adere-se a política de aceitação de conexões permanentes. Como resultado, o controle da política de aceitação está presente de forma automática no sistema. Essa política é um conjunto de regras usadas pelo gerente para aceitar ou recusar uma requisição de PVC por parte de um usuário.

#### 4.7.4. A localização da política de controle no ciclo de vida das SVCs

Durante seu ciclo de vida, uma SVC passa por três fases distintas, durante as quais pode sofrer interferência da política de controle [53]: Estabelecimento, Duração e Desconexão (Figura 4-6).

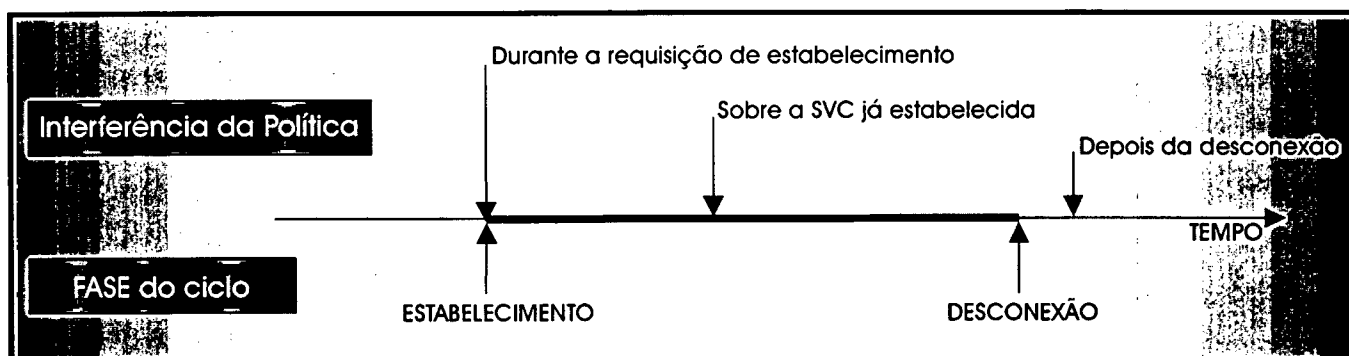


Figura 4-6 - Ciclo de vida da SVC e pontos de interferência

##### 4.7.4.1. Verificação durante o estabelecimento

Quando a política de controle é exercida durante a fase de estabelecimento da conexão, dois pontos devem ser verificados: A verificação normal da função CAC e a política de controle. Devido ao fato que as redes ATM estabelecem conexões SVC sob demanda e essa fase tem uma duração muito pequena, (frações de segundo), ambas as verificações devem ser efetuadas no mesmo intervalo de tempo e, portanto, de forma automatizada. Essa verificação deve ser feita através dos dados de sinalização da rede, e não está implementada na fase inicial da ferramenta proposta neste estudo. A verificação pela CAC, padronizada pelo ATM Forum e implementada pelos comutadores, será mantida sem alterações.

##### 4.7.4.2. Verificação pela CAC:

A função CAC assegura que a nova conexão somente será efetivada se a rede possuir os recursos para acomodar os requisitos da conexão, sem degradar as conexões já existentes. Os recursos que são interessantes nesse sentido são a largura de banda, espaços para as células nos buffers e outros. Os padrões para a CAC já estão

---

estabelecidos pela UNI do ATM Forum e são implementados pelos fabricantes dos comutadores.

Os métodos de CAC são essencialmente métodos de controle preventivo de congestionamento. Estes métodos tornam-se mais importantes a medida que verifica-se que em redes de alta velocidade, métodos de controle dinâmico de congestionamento muitas vezes falham em vista dos tempos de reação envolvidos entre o momento da detecção do congestionamento até o momento efetivo de reação dos usuários frente a ela. Desta forma, a estratégia atual de controle de congestionamento, é no sentido de prevenir o congestionamento do que recuperar a rede de seus transtornos.

Os mecanismos de controle CAC podem ser considerados a primeira e a principal barreira para evitar que uma rede entre em congestionamento. Em tráfegos como CBR, VBR e UBR, não há condições para a atuação de mecanismos dinâmicos e desta forma deverão ser tomados cuidados especiais para prevenir congestionamento.

#### **4.7.4.3. Verificação pela política de controle**

A função da política de controle é dividir os recursos de uma maneira definida para um conjunto de usuários ou sub-redes que estejam competindo pelos recursos. Essa política pode consistir de uma série de regras que definam sob quais circunstâncias (além da simples disponibilidade dos recursos requisitados) a requisição de uma conexão comutada será aceita. Como citado anteriormente, a fase inicial da ferramenta de gerência não implementa tal verificação.

#### **4.7.4.4. Interferindo nas SVCs Existentes**

Interferir nas conexões já existentes é outra maneira de controlar os recursos da rede. Embora de forma mais rudimentar, essa técnica permite ao gerente liberar recursos simplesmente desconectando determinadas conexões. Para fazer isso o gerente (que nesse caso pode ser um humano ou um software de gerência) deve ser capaz de:

- Obter uma visão genérica de todos os SVCs;
- Desconectar uma conexão indesejada;
- Impedir que uma conexão interrompida seja se-estabelecida imediatamente.

Para esse tipo de controle o gerente precisa de um conjunto de ferramentas de software que forneçam as funcionalidades acima. Tal conjunto de funcionalidades está presente na ferramenta de gerência ATM.

A política de interferência está baseada nas seguintes prioridades:

Todas as conexões estabelecidas por interfaces ATM oriundas de máquinas de missão crítica.

Os canais de sinalização ILMI, roteamento e endereço (0,5,16...) devem ser preservados.

Em seguida, as conexões com maior tráfego

Logo após, as mais recentes.

No último nível de prioridade as mais antigas com pouco tráfego.

#### **4.7.4.5. Verificação posterior ao encerramento**

Verificar as conexões após seu encerramento é uma solução adicional para o problema de controle das SVCs. Com essa forma de controle não se pode impedir que algum usuário se apodere de todos os recursos disponíveis, uma vez que a ação é posterior ao encerramento da conexão. Ao invés disso, a verificação posterior nos permite tornar o usuário consciente que sua utilização da rede está sendo monitorada. Essa conscientização pode ser feita através da distribuição de cotas, semanais ou mensais. Um sistema de contabilização de utilização por interface ATM está implementado na ATRM-WTool.

Pode-se, futuramente, implementar a contabilização por usuário, fazendo uso inclusive de MIBs para gerência de contabilização [18]

#### **4.7.5. Áreas de aplicação da política de controle**

As decisões da política de controle devem ser tomadas com base em um conjunto de critérios. Tais critérios podem ser divididos em 4 áreas distintas [53]:

- Para usuários e estações específicas
- Através de créditos remanescentes
- Pelo horário do dia
- Pelo estado atual da rede

---

## 5. Gerência de Redes

---

### 5.1. *Modelo funcional OSI de Gerência de redes*

O padrão OSI definiu 5 áreas funcionais para a gerência de redes, comumente abreviadas como “FCAPS”:

- Gerência de Falhas
- Gerência de Performance
- Gerência de Configuração
- Gerência de Contabilização
- Gerência de Segurança

No estágio atual, a gerência ATM baseada em células OAM cobre apenas as funções de Falhas e Performance, da forma definida pelo modelo funcional OSI [38]. Alguns aspectos de gerência de configuração, falhas e performance são cobertos pelas MIBs ILMI e AToMIB [22] [35] [43] [60]. Permanece uma grande área a ser coberta, particularmente nas funções de configuração e contabilização. Os trabalhos na área de performance/tráfego<sup>6</sup> ainda estão incompletos, e a área de falhas é a mais sedimentada de todas [38], preenchendo alguns dos requisitos para o desenvolvimento da gerência ATM.

### 5.2. *Padronização de Gerência*

A especificação *af-tm-0056.000* [65] descreve os padrões para gerência de tráfego ATM na sua quarta versão (abril de 1996), segundo o ATM Forum. Tal especificação está relacionada com a recomendação I.371 do ITU-T [32]. Algumas diferenças de nomenclatura são encontradas entre os dois órgãos de padronização (Tabela 5-1).

---

<sup>6</sup> Normalmente os padrões do ITU-T utilizam o termo “gerência de performance” e “controle de tráfego” para diferenciar o que o ATM Forum denomina simplesmente de gerência de tráfego [65] [31] [32] - Tabela 5-1.

ATM Forum	ITU-T
Gerência de Tráfego	Gerência de Performance e Controle de Tráfego
Categorias de Serviço	Capacidades de Transferência
Constant bit Rate (CBR)	Deterministic Bit Rate (DBR)
Variable Bit Rate (VBR)	Statistical Bit Rate (SBR)

**Tabela 5-1 - Diferenças de nomenclatura na Gerência de Tráfego**

Uma rede de computadores pode apresentar grandes problemas, a longo prazo, quando não gerenciada convenientemente [38]. Pode-se imaginar a dificuldade de se interconectar máquinas tão diferentes quanto computadores, comutadores, roteadores, se as convenções de gerenciamento (como uso de alarmes, indicadores de performance, estatísticas de tráfego e contabilização) forem diferentes entre cada equipamento ou fabricante. A dificuldade aumenta conforme mais componentes são adicionados a rede, agregando mais funções e mais usuários.

Devido a esse fato, a ISO (*International Standards Organization*) vem trabalhando a algum tempo na padronização da gerência de redes para sistemas abertos (OSI –*Open System Interconnection*). Paralelamente, outra grande força empregada na padronização foi as atividades da Internet, inicialmente através do projeto ARPAnet, assumido posteriormente pelo Departamento de Defesa dos EUA. O trabalho do DARPA (*Defense Advanced Project Research Agency*) levou ao desenvolvimento do protocolo TCP/IP. Nos últimos anos, a *Internet Architecture Board* (IAB) assumiu a liderança na elaboração de padrões para a Internet. Dois padrões de gerência surgiram desses trabalhos:

SNMP (*Simple Network Management Protocol*), um protocolo projetado para atingir soluções de curto prazo para a gerência, evitando as dificuldades da padronização oficial.

CMOT (*Common Management Information Services and Protocol Over TCP/IP*), planejado para solucionar os problemas de gerência a longo prazo, mas que não chegou a ser utilizado em grande escala.

Atualmente, com o advento da Web, novas alternativas de gerência têm sido propostas. Uma delas é o protocolo HMMP (*Hyper-Media Management Protocol*), sob o comando da *Desktop Management Task Force* (DMTF). A DMTF está trabalhando para a definição de padrões para a indústria através de uma nova sintaxe, a *Managed Object Format* (MOF), baseada no *Common Information Model* (CIM). Com uma

abrangência maior em termos de usuários, o próprio protocolo padrão da Web, o HTTP, tem sido utilizado para operações de gerência [17] [6] [28] [9] [42]. Outros padrões, como o CORBA (*Common Object Request Broker Architecture*<sup>7</sup>) estão sendo indicados para gerência na Web [25] [21].

### 5.3. *SNMP*

O protocolo SNMP foi projetado para gerenciar nós na comunidade Internet. Dois documentos definem a informação de gerenciamento: o RFC 1065 'Estrutura da Informação de Gerenciamento' e o RFC 1066- Base de Informação de Gerenciamento. Os dois documentos foram projetados para serem compatíveis tanto com o SNMP quanto com o modelo de gerenciamento OSI.

O modelo atual para gerenciamento de redes baseadas em TCP/IP, está descrito nos seguintes documentos:

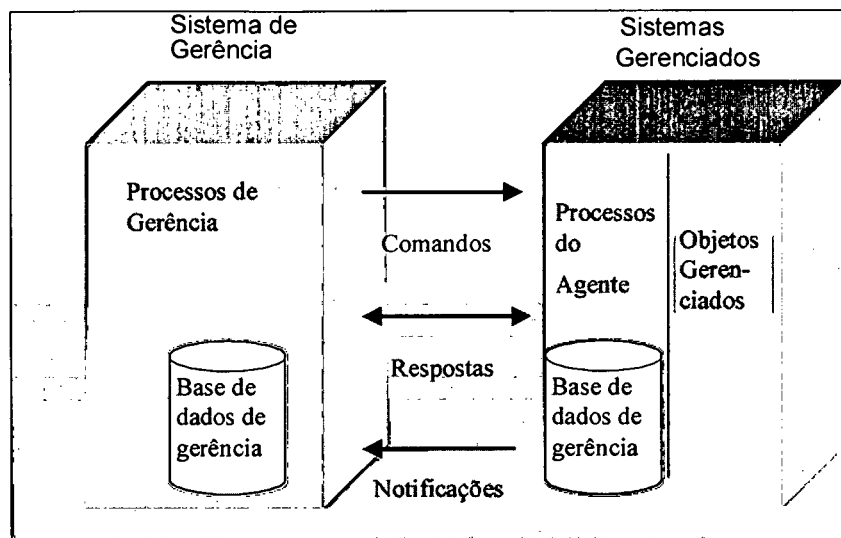
- RFC 1155 [48]- Estrutura e identificação da Informação de Gerenciamento, que descreve como os objetos gerenciados contidos na MIB são definidos;
- RFC 1156 [37] - Base de Informação de Gerenciamento, que descreve os objetos gerenciados definidos na MIB;
- RFC 1157 [10] – *Simple Network Management Protocol* – SNMP (Protocolo Simples de Gerência de Redes), que define o protocolo usado para gerenciar estes objetos.

### 5.4. *Elementos da Arquitetura de Gerência de Redes SNMP*

Um Sistema de Gerência de Redes (NMS – *Network Management System*), denominado “*modelo Gerente/Agente*” consiste dos seguintes elementos (Figura 5-1) [7] [49]:

---

<sup>7</sup> CORBA – *Common Object Request Broker Architecture*- é um padrão definido pelo OMG - *Object Management Group* - que permite a comunicação entre aplicações localizadas em máquinas diferentes e implementadas utilizando diferentes linguagens de programação.



**Figura 5-1 - Modelo de relações Gerente-Agentes**

- Gerente (Ou estação de Gerência [49] – NMS)
- Sistema Gerenciado (Nós gerenciados)
- Base de dados de informações de gerência (MIB)
- Protocolo de rede

O Gerente provê a interface entre o administrador humano e os dispositivos que são gerenciados. O gerente também é responsável pelos processos de gerência da rede. Tais processos compreendem tarefas como medição de tráfego em algum segmento remoto da rede ou gravação da velocidade de transmissão e endereço físico de uma interface de roteador. O Gerente também inclui, tipicamente, uma interface de saída, normalmente gráfica, para apresentar informações de gerência e estatísticas da rede.

O sistema gerenciado, conforme a Figura 5-1, consiste em:

- Processos do agente
- Objetos gerenciados

Os processos do agente realizam algumas das operações de gerência da rede, como a configuração de parâmetros, obtenção do nome e versão do sistema operacional de uma máquina ou alteração do estado operacional de uma interface. Os Objetos Gerenciados consistem de equipamentos dotados de agentes, e incluem estações de trabalho, roteadores, servidores, hubs, comutadores e outros. Associados aos objetos gerenciáveis estão os atributos, que podem ser definidos de três formas:

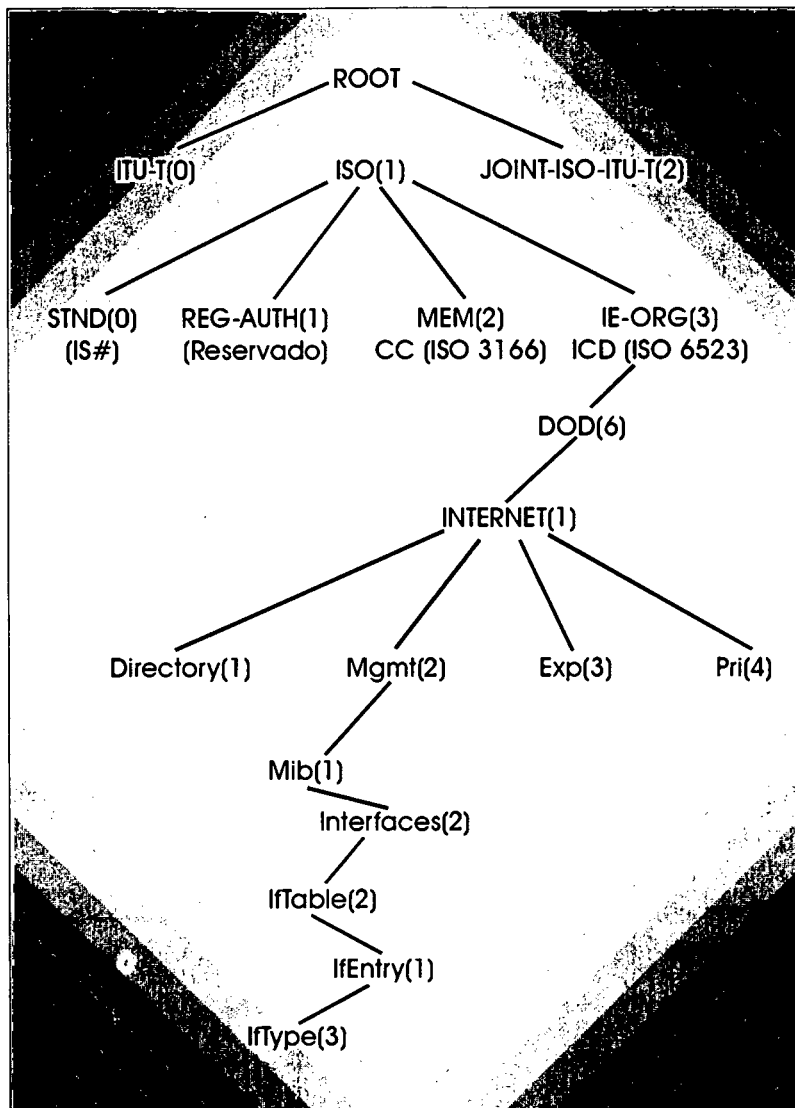


- estaticamente, como a identificação do sistema operacional de uma máquina;
- dinamicamente (como as entradas em uma tabela de rotas);
- continuamente (como as medidas de quantidades de pacotes que são transmitidos sem erro durante determinado período de tempo)

A Base de Informação de Gerência (*Management Information Base* ou MIB) é uma das partes mais importantes de um sistema de gerência [7]. A MIB apresenta uma estrutura hierárquica em árvore, que identifica os objetos gerenciáveis (elementos da rede que podem ser gerenciados). Uma MIB também define de maneira não-ambígua a nomenclatura associada aos objetos gerenciados, representando esses objetos (**Figura 5-2**). Ela pode conter as informações sobre, por exemplo, o número de células descartadas em uma determinada conexão da rede ATM.

As MIBs são associadas com o gerente ou com a estação de gerência (NMS) e com os sistemas gerenciados. Possuindo apenas uma base de dados numérica como estrutura para armazenar e recuperar dados, uma MIB possui uma organização bem definida. Essa organização lógica é denominada *Structure of Management Information - SMI*. A SMI é organizada em forma de “árvore”, começando pela “raiz” e com as ramificações organizando os objetos gerenciáveis em categorias lógicas. As MIBs representam os objetos gerenciáveis como “folhas” nos “galhos” da “árvore” (**Figura 5-2**).

Os atributos associados aos objetos gerenciáveis são necessários para a implementação das funções de gerência, tanto dos agentes quanto dos gerentes [35]. Deve-se notar que existem muitos focos de interesse quando trata-se de gerência de redes ATM. O ATM Forum definiu 5 categorias de interfaces gerenciáveis, denominadas M1, M2, M3, M4 e M5 (seção 6.1).



**Figura 5-2 - Estrutura em árvore das MIBs**

O protocolo SNMP facilita a comunicação entre o dispositivo gerenciado (um agente SNMP localizado em um roteador, por exemplo) e um gerente SNMP (que representa o usuário gerente de redes). A comunicação ocorre entre unidades de dados SNMP (PDUs – ‘*Protocol Data Units*’) [42], [49] e, essencialmente, 4 tipos de operações podem ser executadas entre os agentes e os gerentes :

- *GET*: A instrução *get* simplesmente lê o valor de um atributo do objeto gerenciado.
- *GET NEXT*: Lê o valor do próximo objeto na MIB.
- *SET*: Possibilita alterar valores nos parâmetros ou atributos dos objetos gerenciados.

- *TRAP*: É uma notificação assíncrona emitida pelo agente ao gerente, sobre algum evento ocorrido no objeto gerenciado.

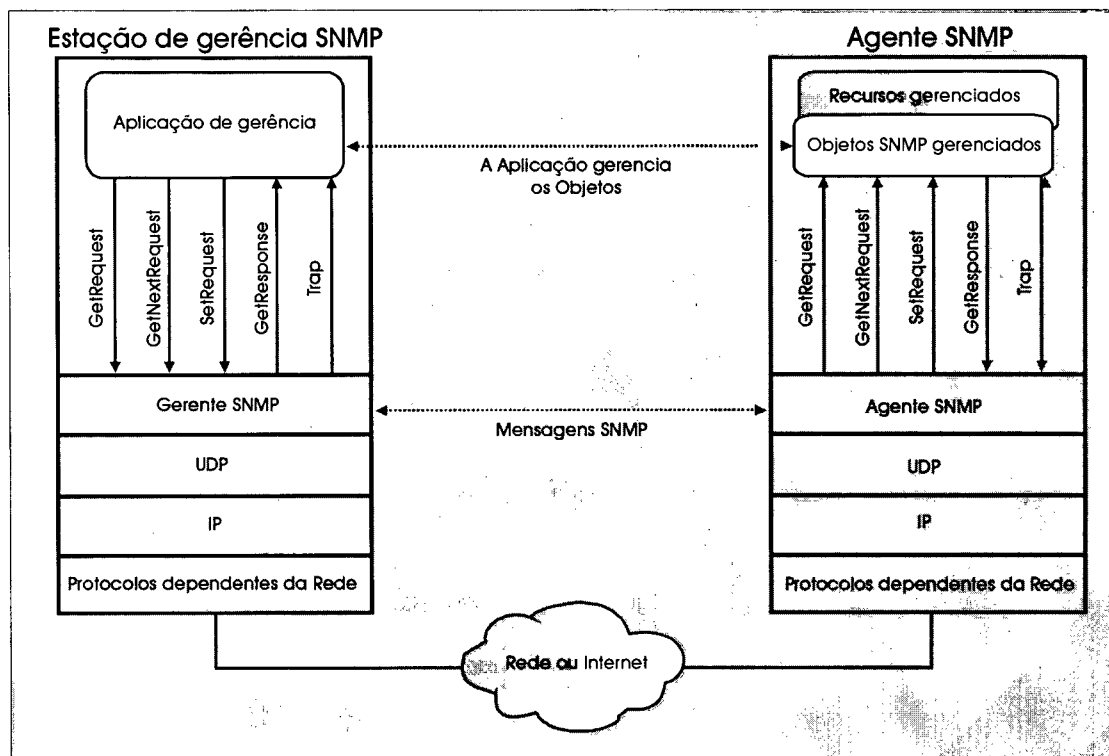


Figura 5-3 - O papel do SNMP ([55])

Com a versão 2 do protocolo SNMP (SNMPv2) [7] [49], outras instruções foram adicionadas ao protocolo.

A implementação de um Agente SNMP pode ser classificada de duas maneiras, dependendo dos objetivos do agente [49]:

- **Fortemente Integrado:** Agentes dessa categoria estão otimizados em função da performance, e desenvolve-se essa característica unindo-se as funções de gerência e as entidades do protocolo em processos lógicos únicos. Mais usados em máquinas com propósitos especiais na rede, como roteadores e comutadores. Cada agente desenvolvido dessa forma possui uma sintonia muito grande com a plataforma na qual está operando. Os agentes não são portáteis dentro de uma mesma linha de produtos de fabricantes diferentes (por exemplo, os comutadores de borda 8271 da IBM, CB1000 da 3Com e Smart Switch 2200 da Cabletron). Muitas vezes, não são portáteis nem dentro de uma mesma linha de um único fabricante (3Com CB 1000 e 3Com CB 7000).
- **Levemente Integrado:** Agentes dessa categoria são desenvolvidos pensando-se em flexibilidade. A implementação do agente é separada das demais entidades do

---

protocolo de gerência. Essa abordagem é utilizada para entidades computacionais de propósito gerais. Usualmente esses agentes são de aplicação genérica. A portabilidade é melhorada estruturando-se a aplicação de forma que possa ser usada no maior número de plataformas possíveis. Normalmente as aplicações Gerentes são desenvolvidas com essa mesma abordagem 'levemente integrada' dos agentes. Usualmente, essas implementações eram feitas na linguagem C, devido principalmente a sua portabilidade. Obviamente, a existência de uma ferramenta multiplataforma como a linguagem Java veio facilitar grandemente a implementação dessa segunda classe de Agentes (ou Gerentes).

## 5.5. Gerência na WEB

As redes atuais geralmente são compostas por um conjunto muito heterogêneo de recursos. É essencial para o usuário a existência de ferramentas com capacidade de controlar essa grande complexidade e variedade. Os protocolos de gerência são a base para conseguir-se controlar diferentes recursos de forma consistente [28].

Existem várias ferramentas no mercado (HP-Open View, Sun NetManager, IBM NetView, Cabletron Spectrum, CA-Unicenter, entre outras) que resolvem, em parte, o problema da gerência. A maioria dessas ferramentas utiliza softwares proprietários e possui uma dependência muito grande das plataformas onde estão instaladas, e da localização geográfica dos equipamentos de gerência. Além disso são ferramentas de custos elevados, e muitas vezes possuem mecanismos complexos de instalação, configuração e operação.

### 5.5.1. Novas Tecnologias de Gerência

Os administradores e gerentes de redes estão expostos a várias tecnologias que se propõe a solucionar todos os problemas de gerência de redes [9] (plataformas abertas, CMIP). Precisa-se, entretanto, saber qual o papel específico da tecnologia na área de gerência de redes. O crescimento acelerado da Internet e das Intranets durante os últimos anos foi promovido pelo surgimento e aceitação imediata das interfaces Web, baseadas em sistemas distribuídos e hipertexto [9]. A simplicidade de apresentar e acessar a informação em sistemas distribuídos e com interface 'amigável' (*user-friendly*) compõe os fatores que promoveram o uso massivo da tecnologia Web.

As empresas estão usando a interface Web para efetuar basicamente dois tipos de gerência [28]:

- A gerência de dispositivos, que trata da configuração de Hubs, Comutadores e Routers.
- A gerência da rede, onde estão envolvidos todos os conceitos da integração agente-gerente (conforme descrito na seção 5.3).

Entretanto, os dois conceitos fundamentais envolvidos com a tecnologia Web, a linguagem HTML (*Hyper Text Markup Language*) e o protocolo HTTP (*Hyper Text Transfer Protocol*) não foram projetados para aplicações cliente-servidor generalizadas.

A função básica inicial da Web é apresentar informações, e não processá-las [28]. Esse quadro deve ser alterado, uma vez que tanto o HTTP e o HTML estão sofrendo evoluções e aperfeiçoamentos constantes. Os browsers Web e os servidores Web também evoluem continuamente, e atualmente possuem a habilidade de executar código.

Conforme aumenta o número das corporações que empregam 'Intranets' para a conectividade das empresas distribuídas geograficamente, os fabricantes de produtos para redes procuram prover os departamentos de informações com vantagens estratégicas na utilização de suas redes para a gerência [17] [6]. Essa estratégia são chamadas de *Web-based Management* (WBM - Gerência baseada na Web), e permitem ao administrador monitorar e manter sua rede usando as mesmas funcionalidades da WWW que tornaram as 'Intranets' ferramentas de comunicação efetivas. Essas funcionalidades permitem aos administradores usarem qualquer browser Web em qualquer ponto da rede, para facilmente configurar, controlar e acessar a rede e seus componentes individuais. Existem atualmente duas abordagens maiores de arquitetura para a WBM (Seções 5.5.4.1 e 5.5.4.2).

### **5.5.2. Gerência de redes na Web**

Provavelmente a primeira experiência comercial de gerência de dispositivos com interface Web seja a do *comutador LocalTalk*, da empresa *Tribe Computer Works* [9]. Foi desenvolvida uma interface simples e fácil de usar, que pode ser seguida como exemplo por todos os dispositivos com poucos parâmetros configuráveis. Em tais dispositivos, o usuário pode controlar as variáveis através do preenchimento de campos em formulários e obter o status a qualquer momento, de forma textual ou gráfica.

Nos anos 80, a gerência de dispositivos era feita geralmente com um terminal de texto. A maior parte desses equipamentos implementava uma pilha TCP simplesmente para permitir o acesso remoto a console. Nos anos 90 o browser Web substitui o terminal texto como modelo. O protocolo HTTP foi bem recebido pelos fabricantes de equipamentos, principalmente naqueles onde não era possível implementar um servidor de telnet, mas HTTP sim.

Da mesma forma que a gerência de equipamentos, a gerência de redes com a interface Web também está se tornando muito popular [17] [6] [9] [42]. Várias empresas têm implementado seus sites de gerência, para uma gama cada vez maior de usuários. A utilização de um browser tem um ganho significativo de custos e treinamento sobre as

plataformas de gerência tradicionais. Tais plataformas têm procurado atualmente migrar para os padrões Web [28]. Essa tendência aponta para a Web como um padrão para a interface das plataformas de gerência.

Alguns problemas de adaptação ao novo paradigma podem ser resolvidos com simplicidade. A interface de gerência padrão VT100 da Cisco, que fez sucesso entre os administradores, foi facilmente adaptada ao novo padrão. A linha de comando poderia ser algo como:

```
'show interface ethernet0'
```

Tal linha de comando tornou-se uma URL:

```
http://routernome/exec/show /interface/ethernet0/
```

Essa abordagem significa que todos que possuam um browser podem agora saber o status de um roteador. Obviamente isso pode ser estendido para resolver vários outros problemas de gerência, criando-se subconjuntos de formulários para funções determinadas. Nos departamentos de suporte, pode-se simplesmente construir uma página com links ao invés de solicitar ao usuário comandos como *'telnet'*.

Pode-se considerar ainda que a WBM é um resultado do crescimento da popularidade das *'Intranets'* [28]. As *'Intranets'* são, na verdade, *'Webs privadas'*. Elas estão sendo usadas cada vez mais intensamente como forma principal de comunicação e compartilhamento de informações dentro das corporações. As redes *Intranet* utilizam o *Transmission Control Protocol* (TCP) e são isoladas do restante da Internet através de *Firewalls* de segurança. São construídas com servidores disponíveis na WEB, usando protocolos relacionados com o HTML. Usuários de uma *Intranet* comunicam-se com os servidores usando a interface dos browsers Web, a partir de qualquer ponto ou plataforma de uma rede. A conectividade é simples, barata e transparente. O acesso a Intranet é feito com segurança através de comunicações *'dial-up'* ou, mais recentemente, através de *'tunelamento'* dentro da Internet, usando o protocolo PPTP (*Point to Point Tunneling Protocol*).

### 5.5.3. Os Benefícios da gerência baseada na Web

A tecnologia WBM mescla as funcionalidades da Web com a gerência de redes para prover os administradores com capacidades além daquelas fornecidas pelas ferramentas tradicionais. Uma vez que os administradores podem, através da WBM, monitorar e controlar toda a rede das empresas a partir de qualquer ponto e com qualquer browser, não existe mais a limitação geográfica para efetuar-se a gerência da

rede. Muitos problemas de interoperabilidade entre as diferentes plataformas são eliminados pela nova estrutura. A interface Web e a operação dos navegadores são paradigmas conhecidos pelos usuários de redes atuais. O custo com treinamento dos gerentes é muito menor, e as informações de *status* podem estar disponíveis para um número bem maior de usuários da rede.

Pode-se dizer que as aplicações de gerência baseadas na Web possuem as seguintes propriedades [17]:

- Diferentes ferramentas possuem a mesma ‘aparência’ (*‘look and feel’*);
- configuração limitada: o próprio sistema localiza os recursos da rede, e limita as opções do usuário, fornecendo valores de configuração *default*;
- ajuda *on-line*: a documentação e outras facilidades estão disponíveis *on-line*;
- suporte ativo: em caso de erro, os sistemas identificam o problema e indicam as possíveis soluções.
- alternativas limitadas: O sistema mostra apenas as alternativas válidas, prevenindo o usuário da utilização errada da ferramenta.

#### **5.5.4. Estratégias de implementação:**

Existem duas abordagens básicas na implementação da WBM. Elas estão evoluindo em paralelo, e não são mutuamente exclusivas.

##### **5.5.4.1. A estratégia ‘proxy’**

Nessa abordagem, um servidor Web é adicionado a uma estação intermediária (a estação ‘*proxy*’) a qual por sua vez comunica-se com os sistemas finais (Figura 5-4). Um usuário em um browser comunica-se com o *proxy* usando o protocolo HTTP, enquanto o *proxy* comunica-se com os sistemas finais usando SNMP [28]. Tipicamente, os fabricantes desenvolvem soluções para utilização com *proxy* adicionando um servidor Web em uma ferramenta de gerência já existente. Isso leva a um aproveitamento das capacidades da ferramenta tradicional, como acesso a banco de dados e ‘*pooling*’ SNMP. A estratégia *proxy* também é denominada ‘tradicional’, uma vez que adota a estrutura já existente na rede.



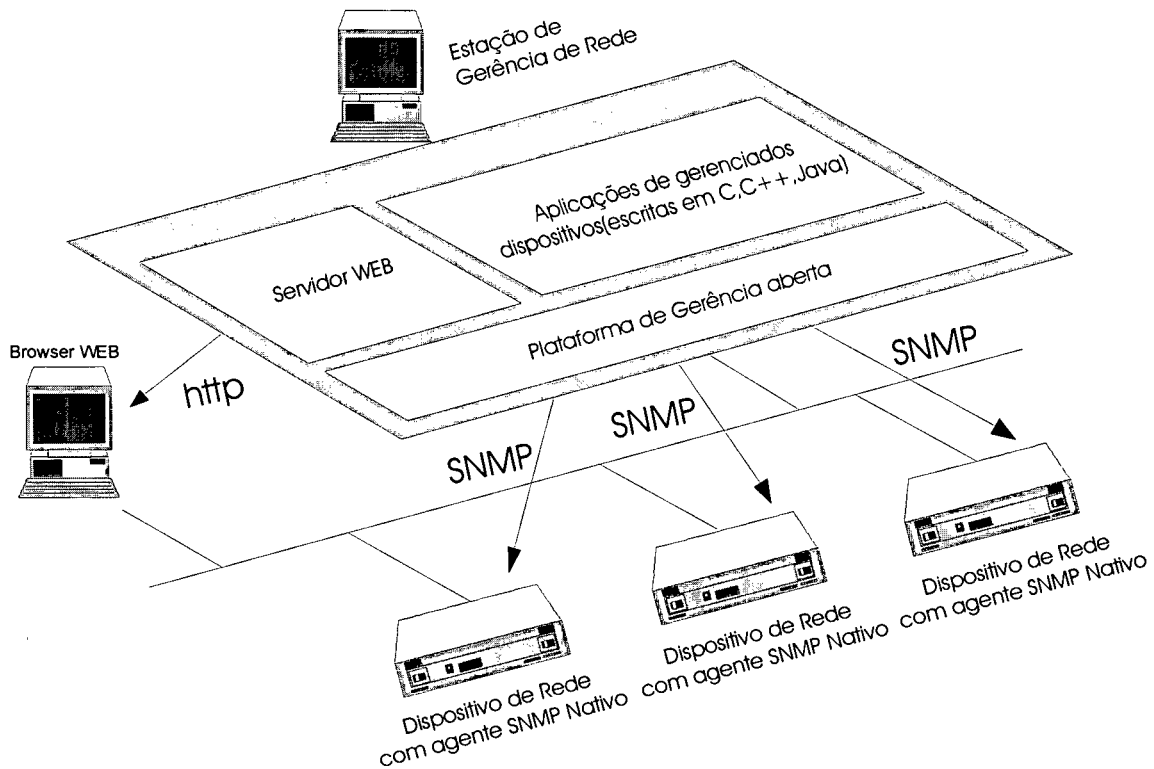


Figura 5-4 – A solução ‘Proxy’ para a WBM

#### 5.5.4.2. A estratégia ‘Embutida’

Nessa segunda solução, o servidor Web é ‘embutido’ no dispositivo gerenciado [28]. Cada dispositivo possui seu próprio endereço Web, e o administrador simplesmente conecta-se ao endereço com um browser para fazer a gerência (Figura 5-5).

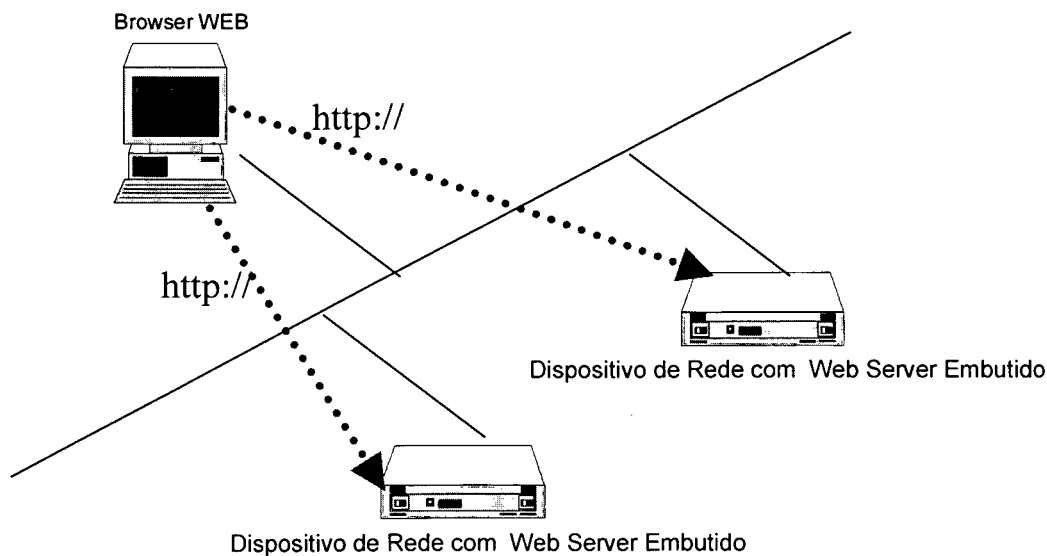


Figura 5-5 - Abordagem 'Embutida' para WBM

A solução com a utilização de um agente proxy preserva todas as vantagens da gerência tradicional, baseada em uma estação de gerência central, somadas com as vantagens da flexibilidade de acesso da Web. Uma vez que o *proxy* comunica-se com todos os dispositivos da rede, ele pode prover uma visão global da rede, tanto dos dispositivos físicos quanto das conexões virtuais, como as LANs lógicas. Além disso, como essa abordagem conserva a comunicação SNMP, ela funciona bem com os dispositivos já existentes na rede, baseados somente no SNMP. Entretanto, necessita de uma estação para operar como proxy.

A abordagem embutida, por outro lado, proporciona interface Web para a gerência dos dispositivos individualmente. Esse tipo de comunicação com os dispositivos de rede é mais fácil de utilizar que as atuais linhas de comando obtidas com o *Telnet* ou com a interface serial.

### 5.6. *Java e SNMP*

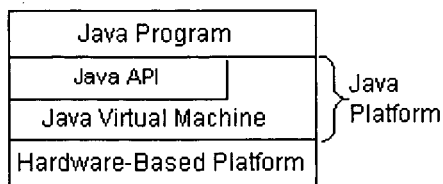
A linguagem de programação Java é uma ferramenta orientada a objetos bastante popular atualmente, com vantagens de ser relativamente simples e possuir bom suporte para computação distribuída, portabilidade extremamente elevada e um grande conjunto de componentes de software já produzidos, os quais fornecem uma ampla gama de capacidades. Essas classes são agrupadas em “pacotes” e distribuídas por uma ampla gama de fornecedores, pesquisadores e estudantes. Normalmente, os pacotes possuem uma habilidade específica sobre um determinado problema computacional, como os pacotes gráficos, matemáticos, estatísticos. Alguns são gratuitas, e suportados por listas e grupos de discussão.

Esse é o caso das classes utilizadas neste estudo. Elas fazem parte do AdventNet Builder, uma ferramenta de desenvolvimento totalmente escrita em java, que possui vários pacotes desenvolvidos para gerência de redes, com capacidades plenas sobre o protocolo SNMP.

A linguagem empregada pode ser utilizada em praticamente qualquer plataforma e browsers de rede, o que significa que tais browsers podem “carregar” e “rodar” programas escritos em Java

Uma das grandes vantagens da linguagem é sua portabilidade. O mesmo *byte-code*, que são os fontes compilados em Java, podem rodar em qualquer Máquina Virtual Java (Java Virtual Machine -JVM). Naturalmente, a Máquina Virtual é dependente de plataforma, uma vez que ela deve estar integrada com os recursos

específicos do sistema. A **Figura 5-6** mostra um programa java (uma aplicação ou applet) que está rodando em uma plataforma Java. Como se pode perceber, a API Java e a máquina virtual isolam o programa das dependências do hardware.



**Figura 5-6 – Java Virtual Machine**

A máquina virtual Java é construída dentro dos browsers, como o Netscape e o Internet Explorer de forma que tais ferramentas sabem como executar os arquivos compilados (.class)

A linguagem Java incentiva a utilização de *applets* ao invés de processos *stand-alone*. Um *applet* é uma instância que roda em alguma máquina local, enquanto seu código e dados residem em outro local qualquer da rede. Um bom exemplo de utilização é o acesso aos servidores de Bancos de Dados. O cliente carrega o applet, que imediatamente passa a fazer suas requisições ao servidor. Ao contrário dos processos *stand-alone*, um *applet* precisa de um “hospedeiro”. Esse papel é desempenhado pelos browsers Web.

Existem algumas empresas que implementaram classes em Java para manipulação do protocolo SNMP (entre elas a Advent – <http://www.adventnet.com> ). As classes implementadas pela Advent foram utilizadas neste trabalho, devido ao fato de serem acessíveis publicamente e possuírem utilização e operação difundida através de listas de discussão. O Software de implementação Advent Builder 3.0 foi utilizado para o desenvolvimento da ferramenta ATM.

As plataformas de gerência, quando desenvolvidas em Java, permitem o acesso ao servidor através de qualquer estação conectada a rede, mesmo um notebook com acesso discado ou com conexão *wireless*. Isso é possível devido ao conceito de ‘máquina virtual’, que roda em qualquer Browser Web. Cada classe escrita em Java é compilada como arquivos de classes separados. Tais arquivos podem ser carregados no browser do cliente em tempo de execução. O *Java Development Kit* (JDK) é a ferramenta básica para se compilar e rodar programas em Java.

Os Applets Java são programas que podem rodar em qualquer Browser Web habilitado para Java, como o Netscape e o Internet Explorer. Por questões de segurança,

---

os browsers oferecem restrições às capacidades dos applets. Atualmente, um applet não consegue comunicação com nenhum host da rede a não ser com o servidor Web de onde foi instanciado. Essas restrições podem ser diminuídas com a validação de assinaturas digitais que comprovem a veracidade das fontes dos applets.

Uma aplicação Java é um programa normal, que não roda no ambiente do browser Web e, sendo assim, sofre somente as restrições normais do sistema operacional.

As classes Java do pacote Advent SNMPv2c permitem a construção tanto de applets quanto de aplicações, e estão divididas em 4 categorias:

- SNMPv2c Variable Classes
- SNMPv2c Communication Classes
- SNMPv2c MIB Related Classes
- Miscellaneous Classes

---

## 6. Gerência de Redes ATM

---

Como em outras áreas de gerência de redes, várias arquiteturas têm sido propostas para gerenciar os diversos planos do modelo de referência B-ISDN definidos pelo ITU-T. Destacam-se aquelas propostas apresentadas pelos organismos internacionais de padronização envolvidos com a tecnologia ATM: OSI, ITU, IETF e ATM Forum. Todas essas arquiteturas possuem pontos comuns, mas existem algumas diferenças de abordagem, abrangência e utilização de protocolos. O presente estudo baseia-se nas especificações do ATM Forum [65] [64], que por sua vez possuem vínculo estreito com as recomendações do ITU-T [31] [32].

O protocolo SNMP continua sendo o preferido para a gerência de redes ATM privadas [35], que são padronizadas pelo ATM Forum. As redes públicas, padronizadas pelo ITU-T, dão preferência para o protocolo CMIP.

Muita discussão sobre a integração de plataformas é encontrada na literatura [50] [38], mas observando-se em detalhes, a gerência de redes permanece como um conjunto de procedimentos isolados, criados para resolução de problemas específicos [46] [17] [6] [15] [39]. As ferramentas e técnicas utilizadas para gerência de tráfego em LANs (*Local Area Networks*) não se aplicam com as mesmas facilidades em tráfegos de WANs (*Wide Area Networks*) [51]. Da mesma forma, conexões que manipulam aplicações sensíveis ao retardo (tráfego de voz em CBR, por exemplo) requerem atenções especiais que simplesmente não se aplicam às conexões assíncronas, independente da escala de abrangência da rede.

Esse tipo de abordagem, fragmentada em função do tipo de tráfego e da área de abrangência da rede, não se aplica na gerência de redes ATM. Isso porque as redes ATM foram projetadas para transportar tipos diversos de tráfego e suportar aplicações variadas, tanto em redes locais como em WANs.

A grande maioria das ferramentas atuais de gerência foi especificada para redes não ATM [2]. Além disso a grande maioria dos padrões de gerência para ATM são recentes e continuam sem ser implementados pelos fabricantes. Essa fase de desenvolvimento das tecnologias leva os fabricantes a desenvolverem extensões proprietárias para seus agentes gerenciáveis. Isso leva a uma situação onde a tentativa de gerenciar equipamentos de diferentes fabricantes pode ser bastante difícil. Sob um

determinado ponto de vista, pode-se dizer que os dispositivos ATM possuem pelo menos agentes rudimentares de gerência [2]. Embora o IETF e o ATM Forum continuem trabalhando para padronizar a gerência das redes ATM, muitos tópicos precisam ser cobertos. Obter-se hoje uma visão do tráfego ATM em uma rede não é tarefa impossível. A questão chave é o que procurar, e onde procurar as variáveis de interesse.

Em muitos aspectos, gerenciar uma rede ATM é como uma gerência tradicional, recaindo nas cinco funções essenciais descritas na seção 5.1 (FCAPS). As diferenças começam nas taxas de transmissão, mais elevadas e diversificadas. Isso significa que quantidades bem maiores de informações serão emitidas para a estação de gerência em um dado intervalo de tempo, possivelmente em grandes rajadas. Deve-se considerar também que os diferentes tipos de tráfego envolvidos pela tecnologia acrescentam mais informações a serem monitoradas. Talvez a principal diferença resida no fato de que, ao gerenciar uma rede ATM, o gerente deve ter bem claro que tratam-se de duas redes distintas: uma física e outra virtual. Essa última tem suas complexidades inerentes, seja pela quantidade de conexões ou pela informação de sinalização que transita pela rede nos diferentes planos do modelo de referência.

As estações de gerência (NMS) também devem ser capazes de suportar grandes volumes de dados. E as aplicações de gerência devem ter capacidade de correlacionar as informações, tornando fácil a detecção dos problemas críticos, sem necessidade de “garimpar” esses dados. A aplicação de gerência deve estar adaptada a topologia da rede, podendo dessa forma classificar os dados pela localização dos comutadores.

Por outro lado, é necessário aos gerentes de uma rede ATM revisarem as condições de uma conexão fim-a-fim. Tal revisão passa pelos conceitos de retardo de transferência das células, congestionamento e descarte. Isso tudo para dezenas de milhares de conexões virtuais. Os gerentes precisam também monitorar e controlar a configuração da infra-estrutura dos comutadores ATM. Atualmente, nenhuma das plataformas de gerências de redes (baseadas em SNMP) oferecem o conjunto de capacidades necessárias para gerência de uma rede ATM de grande escala [4]. Ferramentas como HP OpenView™, IBM NetView™, Cabletron Spectrum™, Sun NetManager™, CA Unicenter™, 3Com Transcend, não possibilitam a gerência das conexões fim-a-fim dos milhares de circuitos virtuais comutados de uma rede ATM através das redes locais ou das WANs.

Os padrões continuam a ser desenvolvidos em muitas áreas do ATM.. A padronização da gerência de redes normalmente é a última fase de um ciclo de vida de uma tecnologia. Isso porque, usualmente, somente depois de instalar a rede, determinar os pontos frágeis, e o que pode ser feito para corrigir eventuais falhas relativas a esses pontos, é que pode-se projetar a filosofia de gerenciamento de uma rede. A gerência da nova tecnologia é complicada pelo fato de que as padronizações do ATM e da B-ISDN estão destinadas a suportar os sistemas legados tanto quanto os novos serviços, funções e aplicações. Além disso, o ATM muda as camadas inferiores da comunicação de dados através do seu novo paradigma, o que torna muitas das verdades do paradigma de orientação a circuito obsoletas, ou pelo menos requerendo revisão substancial.

Com o ATM, o fator multiplicativo que pode ser definido através dos vários canais lógicos ou virtuais em cada interface física é mais de uma ordem de magnitude maior que os meios convencionais de transmissão. Simultaneamente, o ATM inclui serviços de LAN, MAN e WAN com uma extensibilidade futura ainda não definida. Além disso, o ATM introduz novas capacidades, como a conexão ponto a multiponto. Some-se a isso o fato de que uma rede multiserviço exige compatibilidade com sistemas antigos de transmissão, e requer para isso uma capacidade de gerência pelo menos igual a existente nas redes atuais.

Existem vantagens no fato do ATM suportar todos esses tipos de serviços: os Sistemas de Gerência de Rede (Network Management Systems – NMS) baseados em ATM podem prover um nível de gerência sem fronteiras entre as classes de rede (LAN, MAN e WAN), abrangendo a todas em uma única interface de usuário. As interfaces padronizadas de gerência (SNMP, CMIP) com os Elementos de Rede (Network Elements – NE) permitem um alto grau de visibilidade e controle sobre o qual um NMS pode ser desenvolvido.

As novas características de comutação e multiplexação dos equipamentos ATM requerem melhorias nos sistemas de gerência existentes [38]. Os parâmetros dos protocolos adicionais de AAL e superiores também requerem gerência. Monitorar qualquer conexão virtual usando tanto células OAM<sup>8</sup> e/ou medidas de contagem resulta em grandes volumes de dados de gerência. Além disso, devido a sua alta velocidade e pequeno tamanho das células, o controle de congestionamento nas redes ATM apresenta dificuldades não encontradas nas outras redes [57].

---

<sup>8</sup> Células OAM – Operations, Administration and Maintenance - são células destinadas ao gerenciamento da rede.

### 6.1. Módulos de Gerência ATM

Deve-se notar que existem muitos focos de interesse quando trata-se de gerência de redes ATM. O ATM Forum define cinco 'Módulos de Gerência' para caracterizar as diferentes interfaces gerenciáveis de uma rede ATM [35]. Tais interfaces são denominadas de M1, M2, M3, M4 e M5 (Figura 6-1).

As interfaces M1 e M2 referem-se a conexão entre o sistema de gerência implantado na instituição privada usuária da rede um sistema final ATM privado. A interface M3 é a interface de gerência do cliente da rede ATM pública. A interseção das redes pública e privada começa na interface M4, enquanto a interface M5 é responsável pela gerência entre sistemas de provedoras da rede pública.

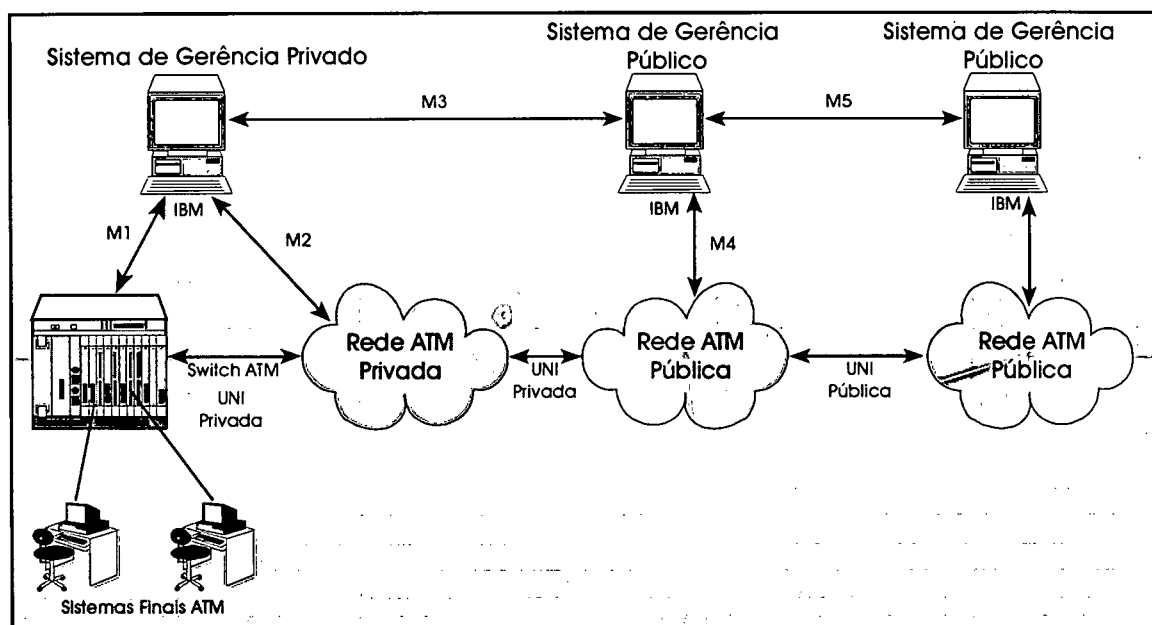


Figura 6-1 - Módulos de Gerência do ATM Forum

O ATM Forum especificou algumas definições de MIBs para essas interfaces, que apresentam-se resumidas na Tabela 6-1.

#### 6.1.1. MIBs das categorias M1 e M2

##### 6.1.1.1. ILMI-MIB: (Integrated Local Management Interface)

Essa interface M1 fornece o status, a configuração e controla a informação para duas interfaces ATM, como entre dois dispositivos ATM. Esses dispositivos podem ser dois comutadores, por exemplo. A MIB ILMI cobre a configuração dos circuitos físicos



---

e virtuais, bem como fornece informações estatísticas e status de performance. Devido a sua importância na gerências dos parâmetros de tráfego, a MIB ILMI será vista em detalhes, na seção 7.4

#### **6.1.1.2.DXI-MIB: (Data Exchange Interface)**

Define rotas para a interface da Unidade de Serviço de Dados (SDU) ATM. A MIB DXI suporta a gerência de configuração e performance da interface DXI, que é basicamente uma interface M1.

#### **6.1.1.3.LANE-MIB: (Lan Emulation)**

Compreende a MIB do cliente Lan Emulation (LEC), a MIB do Lan Emulation Server (LES). A MIB Lane é uma interface de categoria M2. Juntas, elas comportam a MIB-ELAN (LAN Emulada), e possibilitam a gerência de configuração, falha e performance dos clientes e servidores da ELAN.

### **6.1.2. MIBs da categoria M3**

#### **6.1.2.1.MIB M3**

Define os objetos da parte cliente privada de uma rede pública. A MIB M3 é a interface entre a rede pública e a rede privada. O lado da rede privada utiliza o protocolo SNMP, enquanto a porção pública utiliza CMIP. Em consequência, uma função de gerência interna deve ser usada para interoperabilidade.

### **6.1.3. MIBs da interface M4:**

#### **6.1.3.1.M4 NE MIB (Network Element)**

Define os objetos para o domínio da rede e das definições de sub-redes, conexões entre as sub-redes, caracterização de tráfego e estatísticas de tráfego. Cobre as gerências de conexão dos VPs/VCS, gerência de performance e a configuração de interfaces ATM.

#### **6.1.3.2.M4 Network View MIB**

Suporta configuração da rede (provisionamento de sub-redes e links) gerência de conexões, gerência de falhas e configuração de performance (monitoramento de congestionamentos). É uma MIB lógica, e permite o desenvolvimento de MIBs específicas do protocolo.

#### **6.1.3.3.M4 Switched Virtual Circuit (SVC) MIB**

Define os objetos relacionados com os SVCs através dos NEs.

#### **6.1.3.4.ATM AAL MIB**

Define os objetos relativos a Camada de Adaptação ATM, para os NEs da rede pública

#### **6.1.3.5.PNNI MIB (Private Network to Network Interface)**

Define os objetos relacionados a PNNI

#### **6.1.3.6.ATM Inverse Multiplexer**

Define objetos para multiplexação inversa de ATM sobre linhas T1.

### **6.1.4. Outras MIBs:**

#### **6.1.4.1.Interface M5 Network to Network MIB**

Define os objetos para a troca de informações de gerência entre duas provedoras de redes ATM públicas.

#### **6.1.4.2.Test Access MIB**

Permite o controle do acesso remoto aos comutadores, para propósitos de testes.

### **6.1.5. MIBs do IETF (Internet Engineering Task Force)**

#### **6.1.5.1.AToM MIB (RFCs 1695 e 2515) [3]**

Essa é uma das primeiras MIBs definidas para a gerência de redes ATM privadas e dispositivos de redes ATM. Geralmente, é útil para a configuração fim-a-fim e para a gerência de performance. Essa MIB é utilizada no presente trabalho. O ATM Forum contribuiu no desenvolvimento da AToM MIB, para assegurar que os objetos e atributos mais importantes estivessem de acordo com a MIB ILMI. Da mesma forma que a MIB ILMI, a AToM Mib será discutida em detalhes mais adiante neste trabalho (seção 7.2).

<b>Interface do ATM Forum</b>	<b>MIBs Aplicáveis</b>
<b>M1</b>	AToM MIB, LANE MIB, DXI MIB, MIBs proprietárias dos NICs
<b>M2 (SNMP)</b>	AToM MIB, LANE MIB, ILMi MIB, CBS MIB, PNNI MIB, MIBs de transmissão (RFCs 1406 [5], 1407 [14], 1595 [8]), IMA MIB, RMON MIB[66] e MIBs proprietárias.
<b>M2 (CMIP)</b>	M4 Network View MIB, M4 Network Element MIB, M2 SVC MIB, ITU-T SONET MIB e MIBs E1/E3
<b>M3</b>	M3 MIB e AToM MIB
<b>M4 (SNMP)</b>	LANE MIB, ILMi MIB, CES MIB, M4 SNMP MIB, ATM AAL MIB, MIBs de Transmissão transmissão (RFCs 1406 [5], 1407 [14], 1595 [8]), IMA MIB, RMON MIB[66]
<b>M4 (CMIP)</b>	M4 NE MIB, ITU-T I.751 MIB, M4 SVC MIB, ATM AAL MIB, MIBs de Transmissão ITU-T (ITU-T G.704, G.706, G.774, G.826, G.882), Bellcore G.1114 e MIB do Network Management Forum.
<b>M5</b>	ATM Forum NNI MIB e ETSI NA5-2212 Carrier-to-Carrier MIB

Tabela 6-1 - MIBS do Modelo ATM Forum [35]

#### **6.1.5.2. Transmission MIBs**

Essas MIBs definem informações de gerência de falhas, configuração e performance para vários sistemas de transmissão que suportam ATM: RFC 1406, para gerência de linhas T1-E1. RFC 1407, para gerência de linhas T3-E3 e RFC 1595 para gerência SONET<sup>9</sup>

#### **6.1.5.3. RMON ATM MIB (AMON MIB)**

Define os objetos dos dispositivos ATM para teste e análise de performance.

#### **6.1.6. MIBs de outros órgãos de padronização**

Além dos organismos citados (ATM Forum e IETF), outros órgãos de padronização internacional contribuem para a definição de MIBs, principalmente para as provedoras de serviços públicos. Entre essas organizações estão o ITU-T, ANSI, Bellcore.

<sup>9</sup> *Synchronous Optical Network*, um conjunto de padrões do ITU-T para transmissão de sinais digitais em redes ópticas

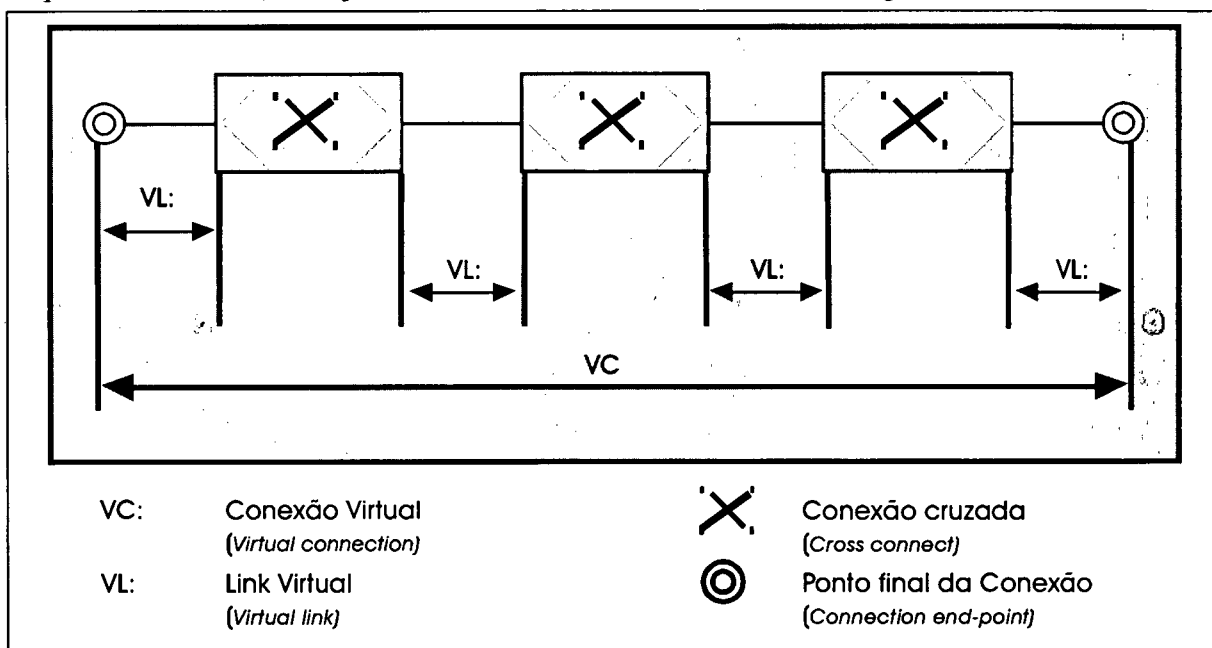
## 6.2. Gerência das Conexões ATM

A gerência das conexões virtuais fim-a-fim é um ponto chave na gerência de redes ATM. As conexões virtuais podem ser divididas em 3 grupos:

- Conexões Virtuais Permanentes (PVCs)
- Conexões Virtuais Semi-Permanentes (*Soft PVC*)
- Conexões Virtuais Comutadas (SVCs)

### 6.2.1. Gerência das Conexões Virtuais Permanentes

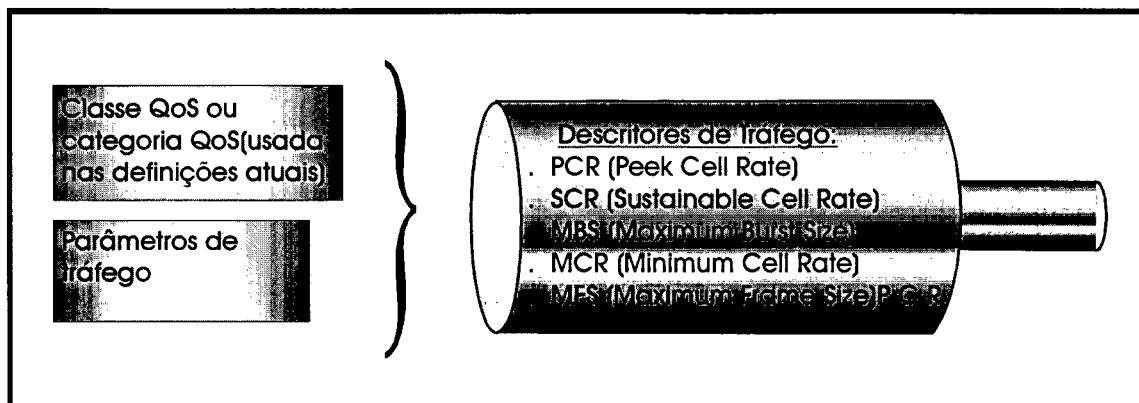
Um único switch ATM ou um único Sistema final ATM não suportam a totalidade de uma conexão virtual im-a-fim (VPC ou VCC). Ao contrário, uma conexão virtual atravessa múltiplos sistemas finais e/ou comutadores, cada qual suportando uma parte da conexão, ou seja um Link Virtual ou VL, conforme a Figura 6-2.



**Figura 6-2 - Conexões virtuais e Links Virtuais**

Isso significa que cada *Sistema final* ATM suporta sua porção final da conexão virtual mais os links nas suas interfaces externas, enquanto cada sistema intermediário (IS – *Intermediate System*), ou comutadores ATM, por onde passa a Conexão Virtual, suportam os múltiplos links virtuais em suas interfaces externas e as conexões cruzadas dos links virtuais que o atravessam. Dessa forma, a gerência de uma conexão virtual fim-a-fim é alcançada pela gerência apropriada de suas várias partes combinadas, como mostrado na Figura 6-2.

Uma conexão virtual é associada com um conjunto de descritores de tráfego, que especificam as características do tráfego, incluindo os parâmetros de tráfego e a classe de QoS, na seguinte relação (Figura 6-3):



**Figura 6-3 - Relação entre as variáveis de Tráfego**

Os links virtuais herdam as características da conexão virtual da qual fazem parte. Os parâmetros de tráfego nas duas direções da conexão podem ser simétricos ou assimétricos. Os descritores de tráfego usados na MIB AToM são consistentes com os usados na MIB ILMI da UNI 3.0 (ATM Forum)

#### **6.2.1.1. Gerência da camada AAL5**

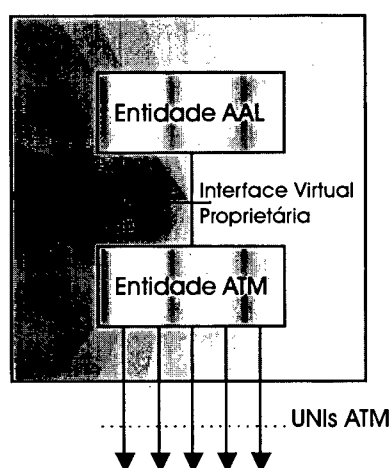
O papel da camada de adaptação ATM é mapear informações dos protocolos de transferência para o ATM. A AToM MIB suporta a gerência da camada AAL5 pela utilização de modelos diferenciados para comutadores e hosts.

### **Gerência da AAL5 em comutadores**

A camada de adaptação 5 é gerenciada em um comutador somente para aquelas conexões virtuais que transportam AAL5 (por exemplo, LANE). Além disso, essas conexões gerenciáveis são aquelas que terminam nas entidades AAL5 internas do comutador, tipicamente para suportar os canais de sinalização.

As VCCs dentro das UNIs ATM transportando AAL5 são comutadas pela "comutador fabric" (aqui entendida como uma entidade ATM) para uma conexão virtual em uma interface proprietária interna associada com os processos da AAL5 (entendidos neste contexto como entidade AAL5). A gerência de performance da AAL5

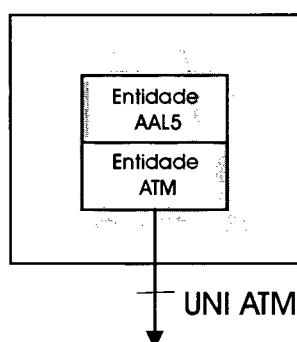
é modelada usando-se a ifTable através de uma interface (pseudo-ATM) interna virtual. A performance da AAL5 por conexão virtual é suportada adicionando-se uma tabela de conexão na MIB ATM. A associação entre os links virtuais na interface ATM é derivada a partir da tabela *VC cross-connect* e da *VC table* na MIB ATM. A gerência da AAL5 no comutador está demonstrada na Figura 6-4.



**Figura 6-4 - Gerência da AAL5 em um Comutador ATM**

#### 6.2.1.2. Gerência da AAL5 em um Host

Gerenciar a AAL5 em um host é uma tarefa direta, uma vez que as VCCs terminam no host no qual a subcamada AAL5 é empilhada diretamente sobre a camada ATM. A gerência da camada AAL5 em um host é demonstrada na Figura 6-5. As interfaces de rede IBM Turboways 25 PCI NIC e 3Com ATMLink 155 PCI NIC, SunATM SBus Adapter e IBM TurboWays 100Mbps Fiber Adapter, utilizadas nos experimentos deste trabalho não implementam a AToM MIB.



**Figura 6-5 - Gerência da Camada AAL5 em um Host**

### 6.3. Gerência de Recursos ATM

Gerência de tráfego é um conjunto de ações na rede que monitoram e controlam o fluxo de tráfego. Ela propicia que cada corrente de tráfego consiga seu nível desejado de largura de banda, e controle sobre a perda de células (*cell loss*), o retardo de células (*cell delay*) e variação do retardo (*delay variation*). A gerência de tráfego suporta níveis de serviço diferenciados. Ela previne que os diferentes fluxos interfiram uns nos outros e previne que cada um dos fluxos consuma mais que seu contrato de uso da rede. Além disso, a gerência de tráfego protege a rede e os sistemas finais (*end systems*) contra o congestionamento.

Em virtude dessas dificuldades e diferenças descritas acima, novos padrões foram definidos. O ITU-T apresentou a especificação I-371 [32] em 1996, referindo-se ao controle de congestionamento e controle de tráfego das redes B-ISDN.

Em abril de 1996 o ATM Forum aprovou a especificação af-tm-0056.000 [65], que refere-se a gerência de tráfego. Esta especificação veio substituir a versão prévia, que se encontrava na especificação UNI 3.1 (*ATM User Network Interface Specification 3.1* [62], de setembro de 1994). Além disso, pode ser considerada como um superconjunto da especificação do ITU-T, pois possui avanços significativos [57].

Como no caso de internets baseadas em IP, o controle de tráfego e congestionamento são vitais para o sucesso das operações nas redes ATM [57].

#### 6.3.1. Contrato de Tráfego

Na gerência de tráfego deve-se considerar que, pelos padrões do ATM-Forum citados acima, essencialmente, existe um contrato de tráfego separado para cada *Virtual Patch Connection* (VPC) ou *Virtual Channel Connection* (VCC). O contrato de tráfego é um acordo entre o usuário e a rede através da User-Network Interface (UNI), a respeito dos seguintes aspectos do fluxo de células ATM:

- A Qualidade de Serviço (QoS) esperada da rede
- Os parâmetros de tráfego que especificam as características do fluxo de células.
- A regra de checagem de conformidade usada para interpretar os parâmetros de tráfego.

#### 6.3.2. A integridade de serviço

A tecnologia ATM consolida a utilização de diversos tipos de tráfego em uma única rede multiserviços. Pela multiplexação estatística dos fluxos de tráfego, uma rede

ATM multiserviços permite que múltiplos circuitos compartilhem os mesmos enlaces físicos (Links) e os mesmos recursos da rede, inclusive a largura de banda [43].

Cada circuito em uma rede ATM possui uma capacidade de largura de banda aparente que é igual a sua taxa de pico. Mas, normalmente, os circuitos não atingem a taxa de pico simultaneamente. Os requerimentos individuais das larguras de banda dos circuitos flutuam ao longo do tempo, produzindo ‘picos e vales’. Como resultado, a soma de todas as taxas de pico de todos os circuitos pode ser maior que a capacidade total da largura de banda do link físico. Uma vez que o ATM combina um grande número de circuitos, os picos e os vales podem se anular, constituindo uma taxa média que produziria uma demanda relativamente consistente e agregada.

A vantagem dessa consolidação de tráfego inclui uma utilização da largura de banda efetiva em termos de custo, uma simplificação das operações da rede, menores custos dos equipamentos e uma oportunidade de expandir a oferta de serviços.

Entretanto, os diferentes tipos de tráfegos de voz, vídeo e dados possuem requerimentos de serviços diferenciados. A migração de serviços oferecidos pelas estruturas de redes tradicionais (Token Ring, Ethernet) para serviços ATM requer uma substituição transparente das redes já existentes. E a substituição transparente requer integridade de serviços, ou seja, a apropriada QoS (Qualidade de Serviço) para os vários tipos de tráfego.

Como resultado, o desafio envolvendo uma rede ATM é alcançar eficiência na utilização do meio enquanto mantém a integridade de serviço. A gerência de tráfego é a chave para encontrar essa meta [43] [47] [57] [20].

### 6.3.3. Contenção

Quando o fluxo de tráfego de múltiplas portas de entrada são destinados a uma única porta de saída, o resultado é a *contenção*. Os comutadores ATM convivem com a contenção durante os tráfegos assíncronos e em rajadas – e mesmo quando tratam de tráfegos de CBR (*Constant Bit Rate*). Os fluxos de tráfego disputam recursos limitados de largura de banda e espaços nos *buffers*. Em muitas arquiteturas, a contenção ocorre tanto dentro do comutador (*switch fabric*) como nas portas de saída.

Contenções sérias, se não resolvidas rapidamente, podem causar sobrecarga no comutador. Sobrecargas prolongadas provocam o congestionamento, uma vez que as aplicações das camadas superiores irão requisitar a retransmissão dos pacotes perdidos. Deste modo, uma rede congestionada não pode garantir os níveis de qualidade de



serviço, e a performance das aplicações das camadas superiores estará seriamente degradada [47] [57].

Para resolver o problema da contenção e fornecer para cada conexão a QoS negociada, um comutador ATM deve possuir funções avançadas de gerência de tráfego [47].

#### **6.3.4. Funções de QoS na Gerência de Tráfego ATM**

A tecnologia ATM é conhecida por suportar uma grande variedade de serviços e aplicações. O controle de tráfego em uma rede ATM é fundamentalmente relacionado com a habilidade da rede em prover Qualidades de Serviços (QoS) para as aplicações da rede. O papel primário da gerência de tráfego é proteger a rede e os sistemas finais (end-systems) contra o congestionamento, possibilitando atingir os objetivos de performance da rede. Um papel adicional é promover o uso eficiente dos recursos da rede [65].

Um conjunto de 5 categorias de serviços é definido na especificação af-tm-0056.000 [65]. Para cada categoria, um conjunto de parâmetros é definido para descrever tanto o tráfego apresentado pela rede quanto a QoS requerida. Além disso, a especificação também define mecanismos de controle de tráfego, os quais a rede pode utilizar para atingir seus objetivos de QoS.

#### **6.3.5. Funções Genéricas de Gerência de Tráfego**

As funções definidas para que a rede alcance suas metas de QoS formam uma estrutura para gerência e controle de tráfego e congestionamento nas redes ATM. Tal estrutura pode ser usada na combinação apropriada para a categoria de serviço a qual se destina. As funções de gerência de tráfego são atualmente as seguintes:

- Connection Admission Control (CAC)
- Feedback Controls
- Usage Parameter Control (UPC)
- Cell Loss Priority Control (CLP)
- Traffic Shaping
- Network Resources Management (NRM)
- Frame Discard
- ABR Flow Control

O ATM Forum deixou a especificação aberta para funções futuras. As seções seguintes descrevem brevemente cada uma das funções genéricas.

#### **6.3.5.1. Connection Admission Control (CAC)**

Essa função é definida como um conjunto de ações tomadas pela rede durante a fase de estabelecimento da conexão (*call set-up*) para determinar se a conexão requisitada deve ser aceita ou rejeitada. O Controle de Admissão de Conexão (CAC) pode ainda re-alocar conexões.

#### **6.3.5.2. Feedback Controls**

São definidos como um conjunto de ações tomadas pela rede e pelos *sistema final ATM* para regular o tráfego submetido a uma determinada conexão ATM, de acordo com o *status* dos elementos da rede.

#### **6.3.5.3. Usage Parameter Control (UPC)**

É definida como o conjunto de ações tomadas pela rede para monitorar e controlar o tráfego, em termos do tráfego oferecido e da validade da conexão ATM, no ponto de acesso dos sistemas finais. Seu maior propósito é proteger os recursos da rede de distorções de comportamento, sejam intencionais ou não. Tais distorções podem comprometer a QoS de conexões já estabelecidas. O UPC (Controle do Parâmetro de Utilização) detecta a violação dos parâmetros negociados, tomando as ações necessárias. Tais ações incluem o descarte de células, por exemplo.

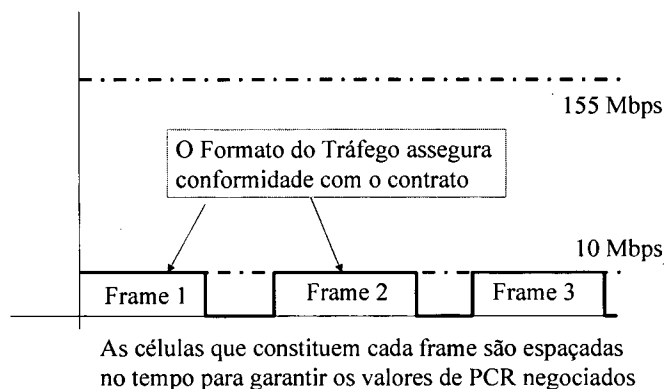
#### **6.3.5.4. Cell Loss Priority Control (CLP)**

Para algumas categorias de serviço o sistema final pode gerar tráfego com marcação de CLP (Prioridade de Perda de Célula) nas células do fluxo. Tal marcação é feita alterando-se o valor do bit CLP no cabeçalho da célula para 1. A rede pode seguir modelos segundo os quais tratará essa marcação como transparente ou significativa. Se tratada como significativa, a rede pode descartar seletivamente as células marcadas com baixa prioridade, para proteger, tanto quanto possível, os objetivos de QoS das células com alta prioridade.

#### **6.3.5.5. Traffic Shaping**

Mecanismos de modelagem do formato do tráfego podem ser usados. O formato do tráfego é um tipo de policiamento de tráfego [57]. Tais mecanismos são importantes para as alterações desejadas, de forma que as características do tráfego correspondam

às negociadas para a conexão. O formato do tráfego é usado para “polir” a forma do fluxo de células e reduzir o descarte [38] [65] [43] - Figura 6-6.



**Figura 6-6 - Formato do tráfego como policiamento**

#### 6.3.5.6. Network Resources Management (NRM)

A arquitetura de serviços permite uma separação lógica das conexões de acordo com as características do serviço. Embora o planejamento de células e provisão de recursos seja dependente da aplicação e do projeto da rede, eles podem ser utilizados para fornecer um isolamento e acesso apropriados aos recursos.

#### 6.3.5.7. Frame Discard

Uma rede congestionada que precise descartar células pode fazê-lo em nível de *frame*, ao invés do nível de células. Em muitos casos tal procedimento é mais efetivo. O conceito de *frame* é designado pelo ATM Forum como “uma unidade de dados da Camada de Adaptação ATM (AAL)” [65]. A rede detecta os limites do *frame* examinando o tipo de SDU no campo de “tipo de Carga” (*payload type*) no cabeçalho da célula ATM. O descarte de *frames* pode ser usado sempre que for possível estabelecer os limites do *frame* através dessa técnica. O descarte de *frames* pode evitar o colapso por congestionamento. Se uma rede suporta o descarte de *frames*, ela pode tratar os dados do usuário como *frame* somente se o usuário permitir esse tratamento.

#### 6.3.5.8. ABR Flow Control

O protocolo de Controle de Fluxo ABR pode ser usado para, adaptativamente, compartilhar a largura de banda entre os usuários participantes das conexões. Um serviço ABR compartilha as capacidades disponíveis na rede [47] [57] [20]. Tais conexões possuem acesso aos recursos não usados pelas conexões CBR/VBR em um determinado momento. Dessa forma, o serviço ABR pode aumentar a utilização da rede sem afetar a QoS das conexões CBR/VBR.

### 6.3.5.9. Outras Funções Genéricas

Na especificação af-tm-0056-000 outras funções genéricas são deixadas em aberto para estudos posteriores.

### 6.3.6. Arquitetura dos serviços ATM

Uma rede ATM pode prover conexões de Canais Virtuais (VC) ou Caminhos virtuais (VP) com níveis distintos de serviço. A arquitetura para serviços propiciada pela camada ATM consiste, segundo a especificação do ATM Forum *Traffic Management Specification* (af-tm-0056.000) [65], de 5 categorias de Serviços. Uma outra categoria (GFR) pode ser citada segundo [24]. Tais serviços são demonstrados na Tabela 6-2.

<b>Categoria</b>	<b>Significado</b>	<b>Tipo de categoria</b>
CBR	<i>Constant Bit Rate</i>	Tempo real
rt-VBR	<i>Real-Time Variable Bit Rate</i>	Tempo Real
nrt-VBR	<i>Not Real-Time Variable Bit Rate</i>	Não tempo real
UBR	<i>Unspecified Bit Rate</i>	Não tempo real
ABR	<i>Avaliable Bit Rate</i>	Não tempo real
GFR	<i>Garanted Frame Rate</i>	

Tabela 6-2 – Categorias de serviço da camada ATM

Essas categorias de serviços referem-se às características de tráfego e aos requerimentos de QoS para o comportamento da rede. Funções como roteamento, CAC (*Conection Admission Control*) e alocações de recursos são estruturados, de uma maneira geral, diferentemente para cada categoria de serviços. As categorias de serviços são distinguidas por serem de *tempo real* e *não tempo real*. Para o tráfego em tempo real existem duas categorias: CBR e rt-VBR. Ambas se distinguem pela presença, no descritor de tráfego, dos parâmetros de SCR (*Sustainable Cell Rate*) em conjunto com o PCR (*Peak Cell Rate*) ou somente o PCR. Três categorias de serviços tratam dos tráfegos “não tempo real”: nrt-VBR, UBR e ABR. Todas as categorias se aplicam indistintamente para VCCs e VPCs. Normalmente o termo conexão é utilizado para designar as conexões virtuais de rotas (VPCs) e de canais (VCCs).

A utilização de Células de Gerência de Recursos (RM-cells) foi designada na especificação af-tm-0056-000 [65] para controlar o fluxo de células em tráfegos ABR. A utilização das células RM para controle dos outros tipos de tráfego não faz parte da especificação. Se tais células estiverem presentes em tais conexões (e sua presença

nesses fluxos é permitida) então elas serão consideradas como parte dos dados do usuário.

### **6.3.7. Definições das Categorias de Serviços**

Cada uma das categorias de serviço possui uma ou mais “Definições de Conformidade”, que são distinguidas através dos valores dos parâmetros de QoS.

#### **6.3.7.1. CBR (Constant Bit Rate)**

Análogo aos serviços de Linhas Privativas. Ele foi planejado para conexões que possuem requisições de largura de banda estática, que permanece continuamente disponível durante toda a conexão. Essa quantidade de largura de banda é caracterizada pelo valor de PCR (*Peak Cell Rate*, ou Taxa de Pico de Células)

#### **6.3.7.2. VBR-rt (Variable Bit Rate – real time)**

Essa categoria é destinada para as aplicações de tempo-real, que impõem fortes restrições de retardo e na variação do retardo, mas não requerem necessariamente uma taxa de transmissão constante ou fixa. Tais conexões são caracterizadas pelo PCR (*Peak Cell Rate*), SCR (*Sustainable Cell Rate* – Taxa de células Sustentável) e MBS (*Maximum Burst Size* – Tamanho máximo da rajada)

É apropriado para fluxos de voz com compressão e supressão de silêncio, bem como algumas aplicações multimídia.

#### **6.3.7.3. VBR-nrt (Variable Bit Rate – not real time)**

Essa categoria, com taxa variável não em tempo real, é apropriada para aplicações em rajadas que necessitam de QoS garantida pela rede. Como a classe anterior (VBR-rt), as conexões VBR-nrt são caracterizadas em termos de PCR, SCR e MBS. Aplicações típicas são reservas de passagens, transações bancárias. Controles de tráfego em redes Frame Relay também podem usar esse serviço.

#### **6.3.7.4. UBR (Unspecified Bit Rate)**

Essa categoria é dirigida para aplicações não de tempo real, em rajadas, que são tolerantes ao retardo e perda de células. O serviço UBR não especifica garantias e, algumas vezes, é citado como ‘serviço do melhor esforço’. Ele pode ser usado para serviços como: transferência de dados, imagem e texto, terminais remotos, *e-mail*, redes *store-and-forward*, interconexão de redes locais, *LAN Emulation*, aplicações de supercomputadores, RPC (*Remote Procedure Call*), sistemas de arquivos distribuídos, processos computacionais de *swapping/paging*.

#### **6.3.7.5.ABR (Avaliable Bit Rate)**

Essa categoria é usada por aplicações que podem tolerar a taxa mínima de células (MCR-*Minimum Cell Rate*), mas estão aptas a consultar a rede para tirar vantagens quando a largura de banda estiver disponível. No estabelecimento dessas conexões, o *sistema final ATM* especifica qual o PCR e o MCR. Um mecanismo de controle de fluxo, suportando vários tipos de consulta à rede, prontamente aloca a largura de banda disponível para a conexão ABR. Exemplos de aplicações para essa categoria são similares aos da categoria anterior (UBR): Compartilhamento de arquivos, impressão remota, redes store-and-forward, interconexão de redes locais, LAN Emulation e outros.

#### **6.3.7.6.Guaranted Frame Rate (GFR)**

O serviço GFR não necessita de aderência a algum protocolo de controle de fluxo. O serviço garante uma largura de banda mínima, mas não tem compromisso algum com o total de perdas quando a aplicação exceder o mínimo garantido. O serviço GFR foi projetado para trabalhar com as PDUs (Protocol Data Units) provenientes das camadas acima da camada de adaptação (por exemplo, AAL-5). A rede objetiva descartar PDUs completas ao invés de descartar células aleatoriamente quando sofre congestionamento. O serviço GFR não faz parte da especificação TM 4.0 do ATM Forum, mas está em desenvolvimento, em conjunto com o ITU-T

### **6.3.8. Definições dos parâmetros de Qualidade de Serviço**

A QoS da camada ATM é medida por um conjunto de parâmetros caracterizando a performance de uma conexão da camada ATM. Esses parâmetros de QoS quantificam a performance da camada ATM fim-a-fim. Na recomendação ITU-T I-356 [31] eles são referidos como “Parâmetros de Performance”.

Seis parâmetros de QoS são identificados na especificação af-tm-0056-000 [65]. Tais parâmetros dizem respeito aos objetivos de performance da rede. Três deles podem ser negociados entre os sistemas finais e as redes.

As categorias de serviço da arquitetura da camada de adaptação ATM (AAL) usam como parâmetros as seguintes variáveis da QoS das conexões [65] [64] [43] [47] [24]:

- *Cell Loss Ratio* (CLR)

- *Máximum Cell Transfer Delay (maxCTD)*
- *Peak-to-Peak Cell Delay Variation (peak-to-peak CDV)*

Não são negociados os seguintes parâmetros de QoS:

- *Cell Error Ratio (CER)*
- *Severely Errored Cell Block Ratio (SECBR)*
- *Cell Misinsertion Rate (CMR).*

Informações detalhadas das QoS da camada AAL podem ser encontradas na Recomendação I.356 do ITU-T [31]. A seguir, são descritos brevemente os parâmetros negociados para as conexões. A negociação destes parâmetros depende da configuração manual (para os PVCs) ou da sinalização (para os SVCs). Os parâmetros são negociados adicionalmente a uma das cinco categorias de serviço da Tabela 6-2.

#### **6.3.8.1. Cell Loss Ratio (CLR)**

A Razão de Perda de Células (CLR) é igual ao número de células perdidas dividido pelo total de células transmitidas em uma conexão. As células podem ser perdidas devido a problemas de funcionamento do comutador, mas usualmente elas são perdidas por descarte explícito do comutador. Tal fato pode ocorrer devido a tentativa de transmissão fora dos parâmetros negociados para determinado fluxo, ou ainda devido ao congestionamento de alguma porta. Segundo Giroux&Ganti [24], a perda de células ocorre devido à sobrecarga das filas dos buffers, originadas por disparos simultâneos de rajadas de conexões diferentes. Além disso, os autores citam ainda falhas nos componentes e comutações de proteção (aquelas que ocorrem para oferecer redundância em caso de falha em algum link) como possíveis fontes de erros. A maneira pela qual um comutador descarta células face a algum congestionamento é crítica para a performance da rede [47]. O comutador pode descartar de uma forma que o congestionamento não seja resolvido, e pode inclusive aumentar o congestionamento. Um algoritmo de descarte baseado em eficiência e seguindo os padrões permite uma melhor performance da rede. O agendamento das *Call Admission Control* e o controle correto das filas podem ser efetivos na perda de células [24]. A razão de perda de células é definida com base em cada conexão (*per VC*) através da relação:

$$\text{CLR} = \frac{\text{Células Perdidas}}{\text{Total de Células Transmitidas}}$$

Nessa equação, considera-se células perdidas:

- O número de células que não chegaram ao destino;
- O número de células que foram recebidas com um cabeçalho inválido;
- O número de células que possuem o conteúdo inválido.

O Total de células transmitidas é o número de células transmitidas em conformidade com o contrato de tráfego durante um período de tempo. O CLR não considera as células que não estiverem em conformidade com os Descritores de Tráfego [31]. O período de tempo para a medida não é padronizado, mas é geralmente considerado como o tempo de duração da conexão. Para as conexões permanentes (PVCs), o tempo de medição é definido pelo operador da rede, e geralmente deve ser grande o suficiente para abranger períodos de pequenos congestionamentos temporários.

A Taxa de erros de células (CLR) pode ser medida tanto para as células com CLP=0 ou para o conjunto de células (CLR=0+1), como determinado pela definição de conformidade.

#### **6.3.8.2. Maximum Cell Transfer Delay (maxCTD)**

O Retardo Máximo na Transferência de Células (maxCTD), analisado nessa seção, é definido como o tempo decorrido entre o 'evento de saída' da origem e o 'evento de entrada' da célula no destino. Tais eventos são definidos pela especificação af-tm-0056.000 [65].

O *Cell Transfer Delay* (CTD) através de uma rede é medido como a soma dos CTD em cada nó participante da rota. O CTD encontrado em um nó pode ser originado, como visto na seção, por fontes internas e externas.

Os *sistema final ATM* que utilizam as categorias de serviço CBR ou VBR-rt (item 6.3.7 acima) fornecem seus requerimentos de CTD negociando um maxCTD com a rede.

#### **6.3.8.3. Peak-to-Peak Cell Delay Variation (peak-to-peak CDV)**

Esse parâmetro pode algumas vezes aparecer na literatura como unicamente *Cell Delay Variation* (CDV) [43], [47]. Ele refere-se ao fato de que, embora as células



possam ser colocadas na rede regularmente espaçadas, vários fatores podem contribuir para que ocorram sobreposições ou espaçamentos (*gaps*) no fluxo das células. A variação do retardo das células é, na maior parte das vezes, um problema dos sistemas finais transmitindo voz, vídeo e aplicações multimídia em conexões CBR ou rt-VBR. Se a rede não pode controlar o CDV, alguns desses serviços de tempo real podem ter suas comunicações distorcidas. Um sistema final usando um serviço CBR ou rt-VBR indica seu requerimento de CDV fim-a-fim negociando um CDV pico-a-pico com a rede. O termo pico-a-pico (*Peak-to-Peak*) refere-se a diferença entre o melhor e o pior caso do *Retardo de Transferência de Células* (CTD – *Cell Transfer Delay*)

#### **6.4. Requerimentos para controle de tráfego e congestionamento em redes ATM**

Os tipos de padrões de tráfego impostos pelas redes ATM, bem como as características de transmissão dessas redes diferem marcadamente das demais redes comutadas. A maioria das redes de comutação de pacotes e Frame-Relay transportam dados em tempo não real. Tipicamente, o tráfego nos circuitos virtuais individuais ou conexões Frame-Relay é em rajadas por natureza, e os sistemas receptores esperam receber o tráfego na forma de rajadas. Como resultado, duas características podem ser observadas nessas redes de dados:

- A rede não precisa saber reproduzir o padrão temporal do tráfego no nó de saída.
- Uma multiplexação estatística simples pode ser usada para acomodar múltiplas conexões lógicas sobre uma mesma interface física entre a rede e o usuário. A taxa média de dados requerida em cada conexão é menor que a taxa de rajada daquela conexão, e a interface usuário-rede (UNI) precisa ser projetada para uma capacidade um pouco maior que a soma das médias das taxas de dados de todas as conexões.

Um grande número de ferramentas tem sido desenvolvido para controlar o congestionamento nas redes comutadas por pacotes. Esse tipo de controle de congestionamento se torna inadequado para as redes ATM. Existem 6 razões para esta inadequação [23]:

- A maior parte deste tráfego não é compatível com um controle de fluxo. Por exemplo, fones de tráfego de voz e vídeo não podem para de gerar células, mesmo quando a rede estiver congestionada.
- O retorno (feedback) é lento devido ao tempo de transmissão das células drasticamente reduzido em comparação com os retardos de propagação através da rede.
- As redes ATM suportam taxas de tráfego diferenciadas, desde poucos Kbps até Gbps. Métodos de controle de congestionamento simples normalmente acabam penalizando uma ou outra extremidade desse espectro.
- Os padrões de tráfego transportados são diversificados (por exemplo, CBR, ABR). Novamente, é difícil para as técnicas de controle de congestionamento tradicionais manipular com precisão essa variedade.
- Aplicações diferentes na rede ATM requerem serviços diferentes dessa rede (por exemplo, serviços sensíveis ao retardo para voz e vídeo e serviços menos sensíveis para dados).
- As velocidades extremamente rápidas na comutação e transmissão tornam as redes ATM mais voláteis em termos de controle de tráfego e congestionamento. Alguma técnica que demore para reagir, mudando as condições da rede, irá produzir flutuações extremas e danosas nas políticas de roteamento e controle de fluxo.

Dois problemas chaves de performance que se relacionam com os pontos acima são os efeitos da latência/velocidade e a variação de retardo da célula (CDV), abordados a seguir.

#### **6.4.1. Efeitos da latência/velocidade**

Considere-se uma transferência de células para uma rede ATM numa taxa de 150 Mbps. Nesta taxa, demora-se  $\approx 3 \times 10^{-6}$  segundos para inserir uma única célula na rede  $[(53 \text{ bytes} \times 8 \text{ bits}) = 424 \text{ bits}] / [(150 \times 10^6 \text{ bps})]$ . O tempo necessário para transmitir uma célula da fonte ao destino depende do número de comutadores ATM intermediários (tempos de retardos internos de cada um) e do tempo de propagação na rota entre as duas pontas. Para efeito de simplificação, deve-se ignorar o retardo interno e considerar-se a propagação na velocidade da luz. Dessa forma, se a fonte e o destino estiverem colocados um na costa leste e outro na oeste dos EUA, o tempo de retardo de propagação é de cerca de  $30 \times 10^{-3}$  segundos. Com esse cenário, e supondo ainda que

uma fonte A queira transmitir um arquivo para um destino B e que o controle de congestionamento implícito esteja sendo utilizado (ou seja, não existem notificações explícitas de congestionamento; as fontes deduzem a presença de congestionamento pela perda de dados). Se a rede descarta uma célula devido a algum congestionamento, B pode retornar uma mensagem de *reject* para A, o qual deve retransmitir a célula descartada, e possivelmente todas as subsequentes. Mas durante o tempo em que a notificação retornava de B para A, a fonte A transmitiu um adicional de  $N$  células, onde:

$$N = \frac{30 \times 10^{-3} \text{ segundos}}{3 \times 10^{-6} \text{ segundos/célula}} = 10^4 \text{ células} = 4.24 \times 10^6 \text{ bits}$$

Ou seja, mais de 4 Megabits de dados serão transmitidos antes que A possa reagir a essa ‘indicação’ de congestionamento. Este cálculo auxilia na compreensão do motivo pelo qual as técnicas que se mostram satisfatórias para controlar as redes mais tradicionais não o são para redes ATM de longa distância. Esse fenômeno pode ser equacionado de forma genérica para diversas redes [56]. Para demonstrar, pode-se considerar um fator de desempenho denominado genericamente “Fator de Performance”, que pode ser calculado dividindo-se o tempo de propagação (P) pelo tempo de inserção (tamanho da célula (L) dividido pela taxa de dados (R)). Equacionando, temos:

$$a = \frac{P}{L/R} = \frac{P \cdot R}{L}$$

Onde:

P=tempo de propagação na rede

L=comprimento da informação (célula/pacote) em bits

R=taxa dos dados em bit/s.

O fator  $a$  pode ser interpretado como o percurso da rede medido em bits (PxR), comparado com o comprimento do pacote em bits (L), ou seja, o número de pacotes (células) que estão na “tubulação” do percurso da rede, desde a fonte até o destino, num dado instante. Pode-se concluir então que quanto menor for  $a$ , menor será o tempo de resposta da rede em relação à realimentação de congestionamento para o usuário. A

Tabela 6-3 mostra alguns valores representativos de  $a$  para diversas redes, considerando pacotes de 53 octetos ( $L=424$  bits).

Rede	R: taxa em Mbit/s	P: tempo de propagação em ms	$a$
LAN Ethernet	10	0,005	0,1
Anel de fibra - FDDI	100	0,01	2,4
Rede MAN	45	0,1	10,6
Rede de Pacotes WAN	0,064	15	2,1
Enlace de satélite	0,064	270	38
Frame Relay WAN (T1)	1,544	15	53
ATM WAN	155	15	5.300
ATM WAN	622	15	21.200

Tabela 6-3 - Valores do fator  $a$  em diversas redes

A grande variação de  $a$ , (mais de 5 ordens de grandeza), da mesma forma que o cálculo aproximado do tempo de resposta demonstrado anteriormente, explica porque as técnicas de controle tradicionais, satisfatórias em outros tipos de redes, falham em redes de longa distância baseadas em ATM.

#### 6.4.2. Cell Delay Variation

O CDV, ou “variação de retardo da célula” é um parâmetro importante principalmente para as aplicações que exigem fluxos constantes. Esse é o caso de sinais de vídeo e voz digitalizados, que podem ser transmitidos em uma rede ATM como um fluxo de células. Especialmente para voz, uma necessidade básica é que o retardo ao longo da rede seja pequeno. Esse tipo de sinal possui outra exigência, normalmente conflitante com a primeira. A variação estatística do retardo (jitter) também deve ser baixa, ou seja, a taxa de transmissão de células da fonte deve ser constante e coincidir com a taxa de chegada ao destino. É inevitável que exista variabilidade na taxa de transmissão das células, devido a efeitos tanto na rede como na fonte de emissão. Em uma primeira abordagem, pode-se considerar as ações do sistema destino para diminuir os efeitos das variações.

Um procedimento genérico para atingir-se uma taxa constante de bits está ilustrado na Figura 6-7.

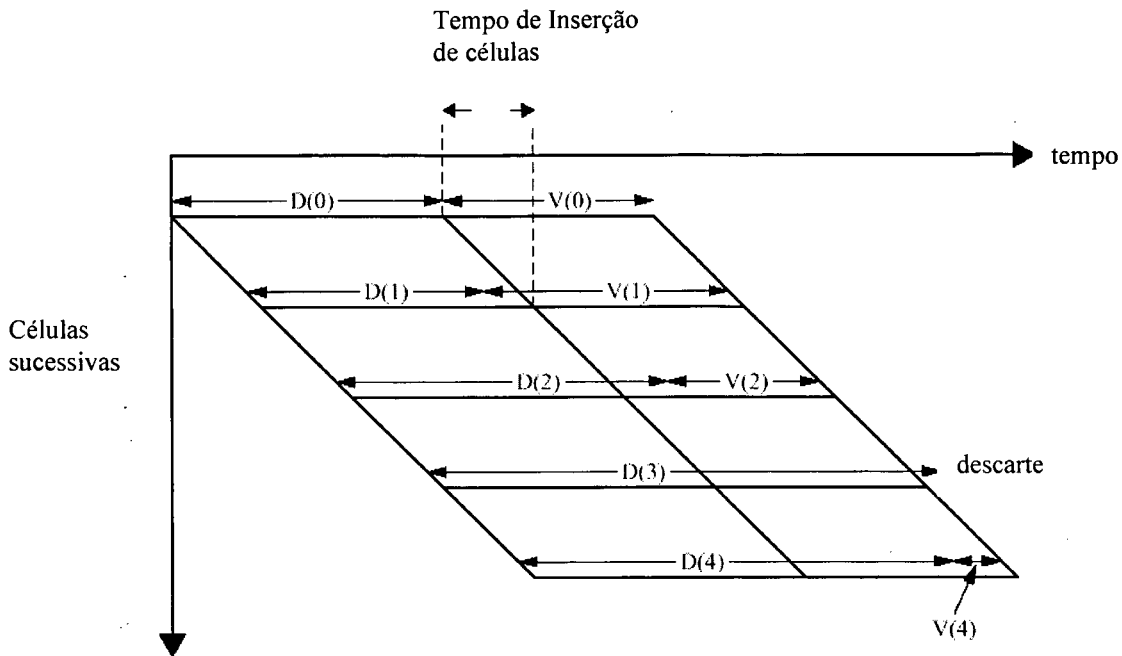


Figura 6-7 - Tempo de re-montagem de células em CBR

Seja  $D(i)$  o retardo fim-a-fim experimentado pela célula  $(i)$ . O sistema de destino não sabe a quantidade exata deste retardo. Não existe marcação de tempo nos bits do cabeçalho da célula. Mesmo que houvesse, seria impossível manter perfeitamente sincronizados os relógios de origem e destino de um fluxo [57]. Quando a primeira célula de uma conexão chega no tempo  $t(0)$ , o sistema de destino retarda a célula em um total de  $V(0)$  antes de entregá-la a aplicação, nas camadas superiores.  $V(0)$  é uma estimativa do total da variação do retardo que esta aplicação pode suportar, ou seja, do valor esperado para o retardo introduzido pela rede.

As células subsequentes sofrem retardos até que sejam entregues para camada de aplicação na taxa de  $R$  células por segundo. O tempo entre a entrega das células para a aplicação de destino (o tempo entre o início da entrega de uma célula e o início da entrega da célula seguinte) pode ser representado por:

$$\delta = 1/R$$

Para alcançar um retardo constante, a célula seguinte deve sofrer um retardo que satisfaça a equação:

$$t(1) + V(1) = t(0) + V(0) + \delta$$

ou:

$$V(1) = V(0) - [t(1) - (t(0) + \delta)]$$

Generalizando:

$$V(i) = V(i-1) - [t(i) - (t(i-1) + \delta)]$$

Quando o valor de  $V(i)$  for negativo, a célula deve ser descartada. O resultado é que os dados são entregues nas camadas superiores em uma taxa de bits constante (CBR), com variações ocasionais devido ao descarte de células.

O valor do retardo inicial  $V(0)$ , o qual é também o retardo médio aplicado a todas as células, é função de uma previsão da variação do retardo da rede. Para minimizar esse retardo, o usuário pode solicitar uma variação do retardo mínima da provedora do serviço. Isso levaria a um compromisso: a variação pode ser reduzida aumentando-se a taxa de emissão da fonte, em relação a carga total e ao mesmo tempo, aumentando os recursos disponíveis nos circuitos virtuais da rede.

## Fontes de retardo

Várias fontes de retardo podem ser apontadas em uma rede ATM. Em uma classificação simples, as fontes podem ser divididas em dois grupos: Fontes de Retardos Internos e Fontes de Retardos Externos [24]. Alguns conceitos referentes a esses dois grupos serão analisados a seguir. Embora em uma visão menos detalhista alguns autores considerem negligenciável o retardo do processamento das células individualmente [57], esse parâmetro deve ser considerado quando níveis mais exigentes de QoS estiverem sendo utilizados [24].

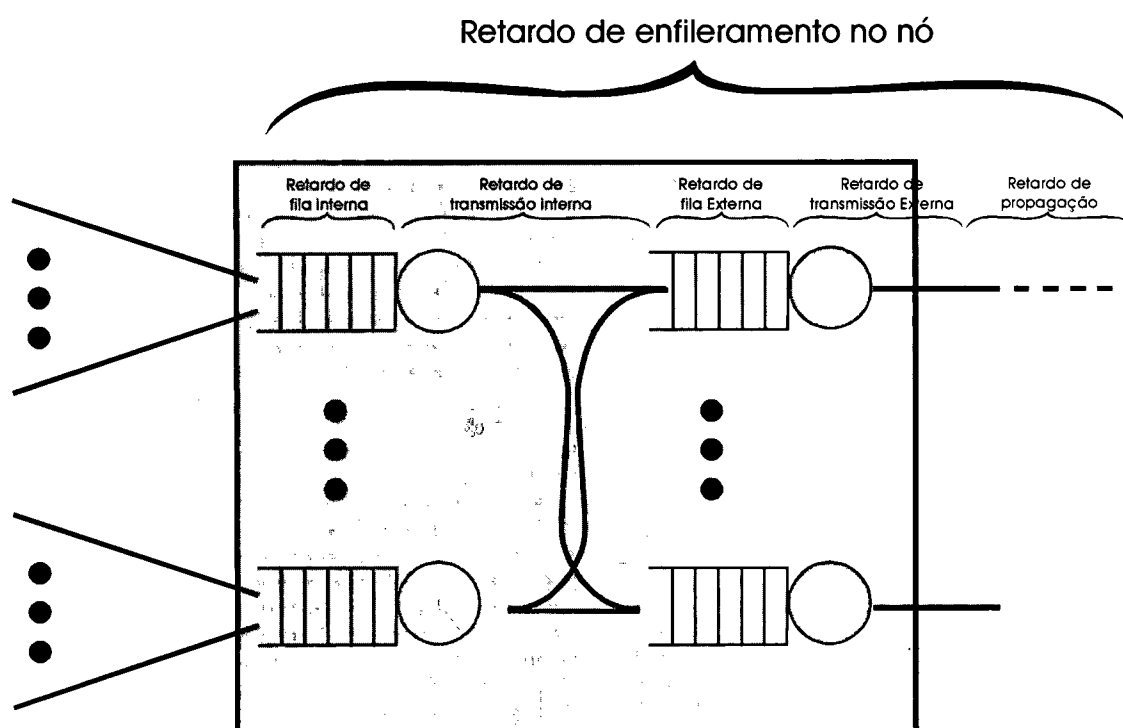


Figura 6-8 – Componentes do *Cell Transfer Delay*

### 6.4.3. *Internal Queuing and Transmission Delay:*

É o tempo requerido para enfileirar e transmitir os bits nos links internos de um nó. A distribuição dos retardos de enfileiramento variam conforme o tipo de carga da célula e do tipo de algoritmo de escalonamento implementado no comutador. A arquitetura do comutador pode necessitar de zero a vários estágios de enfileiramento, com taxas de links internos potencialmente diferentes [13].

#### 6.4.4. *External Queuing and Transmission Delay:*

É o tempo requerido para enfileirar e transmitir os bits em uma interface externa (ou link de saída do comutador, conforme a **Figura 6-8**. A distribuição do retardo pelo enfileiramento das células varia em função da carga e do algoritmo de escalonamento utilizado no ponto de enfileiramento.

#### 6.4.5. *Propagation Delay:*

É o tempo necessário para os bits propagarem-se através do meio físico. Um fator chave no retardo de propagação é o tempo consumido por um sinal entre dois nós. Dependendo do meio utilizado, os sinais eletromagnéticos propagam-se com uma velocidade de 0.2 a 0.6 vezes a velocidade da luz. Esse fator pode se tornar significativo em distâncias muito grandes. O retardo de propagação é calculado como:

$$\text{Propagation Delay} = \frac{\text{Distância}}{(\text{fator de propagação} \times \text{Velocidade da luz})}$$

Onde  $0.2 < \text{fator de propagação} < 0.6$ .

#### 6.4.6. **Processing Delay**

O retardo de processamento representa todos os retardos não negligenciáveis necessários ao processamento das células (por exemplo, análise do cabeçalho).

#### 6.4.7. **Fontes de Retardos Internos**

Os retardos ditos internos são aqueles ocorridos devido a:

- Existência de um ou mais pontos de enfileiramento;
- Comutação;
- Processamento;
- Link de transmissão interno.

#### 6.4.8. **Fontes de Retardos externos**

Os retardos externos são originados pelas filas externas e pelos retardos de transmissão, independentes da existência dos nós. Outros retardos de processamento podem ser adicionados a estes. Quando mede-se o retardo no link, o retardo de propagação também deve ser adicionado ao CTD.



---

## 7. Análise e Implementação das MIBs para gerência ATM

---

Esta seção apresenta um estudo detalhado das MIBs de gerência relacionadas com os parâmetros de tráfego e recursos da rede, que são o objeto deste trabalho. As principais MIBs estudadas foram a MIB do ATM Forum definida no ILMI [64], a AToM MIB [3] [60], a MIB II [39] e a MIB proprietária da IBM [30] para seu comutador central CPSW 8265.

### 7.1. MIB II (RFC 1213)

A aplicação da MIB II para as redes ATM está descrita em várias publicações [3] [60] [45] [11] [11] [52]. A seguir, se irão analisar os grupos relevantes para este estudo.

#### Grupo System

A primeira variável de interesse nesse grupo é a *sysUpTime*, que retorna, em milissegundos, o tempo desde a última reinicialização do sistema. A variável *sysName* será utilizada na tabela *core\_sw* da base de dados, e servirá como chave primária para identificação do equipamento.

Para o propósito do objeto *sysServices*, que determina o conjunto de serviços oferecido pelo sistema, os comutadores e a rede ATM retornam um valor igual a 2. Esse valor é uma soma, que inicia em zero. Então, para cada camada L para a qual o sistema executa transações,  $2^{(L-1)}$  é adicionado a soma. Nesse caso,  $2^{(2-1)}$ . Para um roteador, a variável retornaria 4 ( $2^{(3-1)}$ ). Isso significa que, sob este prisma, um sistema ATM é um sistema do nível de enlace de dados pelo modelo OSI. Existe alguma discussão na literatura a esse respeito, pois a camada ATM não é uma camada de enlace no sentido clássico (protocolos de enquadramento e transferência entre duas máquinas conectadas através de um mesmo meio físico). Tais protocolos são protocolos de um único *hop*, e não tratam das conexões, porque a comutação e o roteamento são tratados na camada de nível 3, a camada de rede, que é a primeira camada *multihop* do modelo de referência [58]. A camada ATM trabalha com movimentação de células da origem para o destino, envolvendo-se com algoritmos de roteamento e endereçamento, funções típicas da camada de rede, no nível 3.

## Grupo Interfaces

O grupo interfaces da MIB II define objetos genéricos para gerência de interfaces. Neste estudo, se irá considerar as extensões específicas para gerência de interfaces ATM, como citado em [60]. Cada interface deve ser considerada como uma camada ATM global, ou seja, representando todas as conexões virtuais (PVCs, e SVCs).

Variável *ifNumber*: É a primeira variável de interesse nesse grupo, pois representa número de interfaces (portas), independente do seu estado (*Up*, *Down*). Essa variável é utilizada pela aplicação ATM para controlar os laços *for - next* no momento da coleta de dados para as portas.

Variável *ifIndex*: Cada entrada em *ifEntry* representa uma porta ATM. Em um comutador CPSW, tem-se os retornos demonstrados na Figura 7-1. Deve-se observar que as primeiras 6 interfaces não representam portas físicas ATM. Os retornos definidos pela variável *ifType* descrevem, a seguir, a representação de cada uma dessas interfaces.

Variável *ifType*: Os valores significativos de retorno para essa variável são representados na Tabela 7-1. Para interfaces ATM, o valor de *ifType* é igual a 37.

Variável *ifOperStatus*: Assume o valor 2 (down) caso a camada ATM esteja nesse estado operacional.

Variável *ifInOctets*: O número de octetos recebidos pela interface, ou seja, o número de células multiplicado por 53.

Variável *ifOutOctets*: O número de octetos transmitidos pela interface, ou seja, o número de células multiplicado por 53.

IfType	Descrição
37	ATM
39	SONET / SDH
49	AAL5
50	SONET Path
53	Proprietária virtual
59	ATM LANE

**Tabela 7-1 - Tipos de interfaces significativas para o ATM**

SNMP Table		
ifIndex	ifDescr	ifType
2	rs232 interface IBM 8265, Part Num: 02L3099, EC level: F12445	33
3	AAL virtual interface IBM 8265, Part Num: 02L3099, EC level: F12445	49
4	LAN emulation ethernet virtual interface IBM 8265, Part Num: 02L3099, EC level: F12445	59
5	LAN emulation 802.5 virtual interface IBM 8265, Part Num: 02L3099, EC level: F12445	60
6	Ethernet interface IBM 8265, Part Num: 02L3099, EC level: F12445	iso88023-csmacd (7)
101	ATM interface 622 Mbits IBM 8265, Part Num: 02L3432, EC level: F12516	37
201	ATM interface 622 Mbits IBM 8265, Part Num: 02L3432, EC level: F12516	37
401	ATM interface 155 Mbits IBM 8265, Part Num: 02L3243, EC level: F12448	37
402	ATM interface 155 Mbits IBM 8265, Part Num: 02L3243, EC level: F12448	37
403	ATM interface 155 Mbits IBM 8265, Part Num: 02L3243, EC level: F12448	37
404	ATM interface 155 Mbits IBM 8265, Part Num: 02L3243, EC level: F12448	37
501	ATM interface 100 Mbits IBM 8265, Part Num: 58G9611, EC level: C38844	37

Figura 7-1 - *ifTable* parcial para um comutador IBM CPSW 8265

Variável *ifInErrors*: O número de células descartadas devido a um erro incorrigível no HEC (*Header Check Error*)

Variável *ifInUnknowProtos*: O número de células recebidas e descartadas durante a validação do cabeçalho da célula, incluindo células com valores de vpi/vci inválidos. Células com PTI não definido e descartadas são contadas também nessa variável.

Variável *ifOutErrors*: Para interfaces de pacotes (células) representa o número de pacotes (células) que não foram transmitidas devido a algum tipo de erro.

Até o momento pode-se observar que as taxas de erros nos comutadores centrais da redeUFSC são nulas para todas as variáveis da MIB II analisadas. O fluxo de células nas portas é tipicamente em rajadas, como demonstra a Figura 7-2

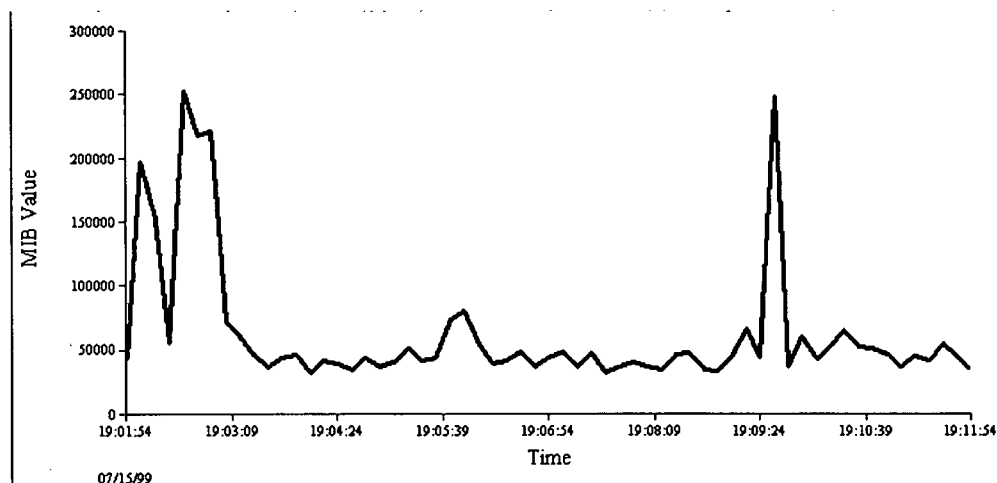


Figura 7-2 - Exemplo de amostragem para *ifInOctets* na interface 101 do comutador CPSW\_NPD

## 7.2. AToM MIB (IETF – RFC 1695/2515)

O RFC 1695 (Definitions of Managed Objects for ATM Versin 8.0) especifica uma MIB para gerência de redes ATM. Como os trabalhos iniciaram em 1993 e foram publicados como padrão ainda em agosto de 1994 [45] , a grande maioria dos comutadores ATM implementa essa MIB, que possui o status “*proposed standart*<sup>10</sup>” pelo IETF. O RFC 2515, que está sendo elaborado para substituir o RFC 1695, possui atualmente o status de “*proposed standart*” e já tem sido implementado por alguns fabricantes, como é o caso da IBM, através do seu modelo 8265 [59]. A principal diferença entre os dois RFCs é o enfoque gerencial: Enquanto a primeira versão da MIB centrava a gerência nos canais virtuais permanentes (VPCs), a segunda versão preocupa-se também com os links comutados (SVCs). O mesmo grupo de trabalho do IETF é o responsável pela MIB de gerência do SONET. A sigla AToM atualmente é a soma de duas partes: ATM e o “o” minúsculo, representando o SONET. A preocupação geral do grupo de trabalho foi manter a MIB peque e eficiente. Embora não seja efetivamente pequena, a MIB é eficiente na gerência, dada a complexidade da gerência ATM.

A AToM MIB define objetos para gerenciar:

- interfaces ATM
- links virtuais
- conexões cruzadas (*cross-connects*)
- entidades e conexões AAL5 suportadas por
- Hosts ATM
- Comutadores ATM
- Redes ATM

A AToM MIB Foi projetada para ser compatível semanticamente com as versões 1 e 2 da SMI, e portanto pode ser utilizada por aplicações de gerência que utilizem tanto o SNMPv1 quanto SNMPv2. Os objetos gerenciáveis da AToM MIB podem ser utilizados também na gerência de serviços. Por exemplo, esta MIB foi especificada para representar uma combinação de comutadores, ou seja, uma rede ATM,

---

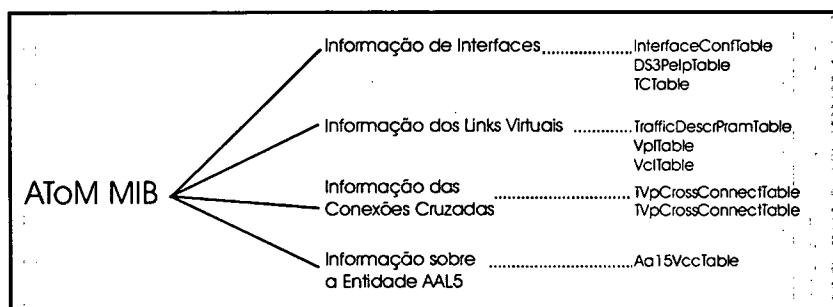
<sup>10</sup> Dentro do sistema de padronização do IETF-IESG, um documento deve percorrer 4 estágios: Experimental, Proposed Standart, Draft Standart e finalmente, Standart.

para gerenciar a interface M3 da especificação do ATM Forum (Customer Network Management).

O papel primário da AToM MIB é gerenciar conexões ATM permanentes, incluindo rotas (PVPCs) e canais (PVCCs). Embora nessa versão as conexões comutadas (SVPCs e SVCCs) estejam representadas, uma gerência completa só foi possível com a definição da segunda versão, como descrito no RFC 2515:

### Estrutura da AToM MIB

A estrutura da AToM MIB está demonstrada na Figura 7-3.



**Figura 7-3 - Estrutura da AToM MIB**

### Variáveis da AToM MIB (ATM MIB 2)

As variáveis da ATM MIB 2 que serão monitoradas pela aplicação ATM estão descritas nos parágrafos seguintes.

Variável *atmInterfaceMaxVpcs*: O número máximo de percursos virtuais (VPCs (PVPCs e SVPCs) ) suportados por essa interface. Em uma interface ATM UNI, a variação possível fica entre 0 e 256. Essa variável será utilizada como parâmetro para o disparo de traps sempre que os valores atingirem mais que 90 % do máximo. Atualmente, o único percurso virtual que está sendo utilizado pelo backbone da redeUFSC é o de índice 0 (zero). Pode-se configurar manualmente ou através de aplicações de gerência, caminho diferentes. Todas as conexões de sinalização estudadas (aquelas utilizadas pelo protocolo ILMI (VCI=16 ) e as conexões da camada de Sinalização da AAL (SAAL – VCI=5) utilizam o VPI= 0, conforme pode ser encontrado na literatura [45].

Objeto *atmInterfaceMaxVccs*: O número máximo de VCCs (PVCCs e SVCCs) suportados pela interface. No ambiente em estudo, essa variável assume um significado

maior que a anterior, pois existem mais canais virtuais sendo utilizados dinamicamente. Para o comutador IBM CPSW, esse número é igual a 1024 conexões. A interface correspondente aos servidores MSS, normalmente instalados no slot 7 de cada chassi é a interface que apresenta o maior número de conexões, chegando no momento a aproximadamente 770 para o comutador CPSW\_NPD. Nota-se que para cada cliente adicionado em um LES, 5 conexões novas são adicionadas na interface do servidor. Outro fator observado com respeito a essa variável refere-se aos comutadores de centro CB 7000 da 3Com. O número máximo de conexões retornado pelo agente foi de 4095 para cada interface, mas o número de conexões estabelecidas chega a 13448 em qualquer interface amostrada. Como essa homogeneidade no número de conexões seria uma coincidência inaceitável, e devido ao fato de as conexões estabelecidas ultrapassarem o máximo permitido, se descartou a possibilidade de coletar os dados neste equipamento.

Objeto *atmInterfaceConfVccs*: Retorna o número de VCCs (PVCC, Soft PVCC e SVCC) em uso atualmente nesta interface ATM. Ela inclui o número de conexões permanentes configuradas e as conexões comutadas que estão atualmente estabelecidas na interface.

Objeto *atmInterfaceMyNeighborIpAddress*: O endereço IP do sistema conectado a essa porta, na outra extremidade do meio físico. É o endereço pelo qual a estação de gerência pode mandar PDUs (*Protocol Data Units*) SNMP.

### **7.3.A MIB proprietária IBM ATM-SWITCHING-NODE-MIB v. 4.1.12**

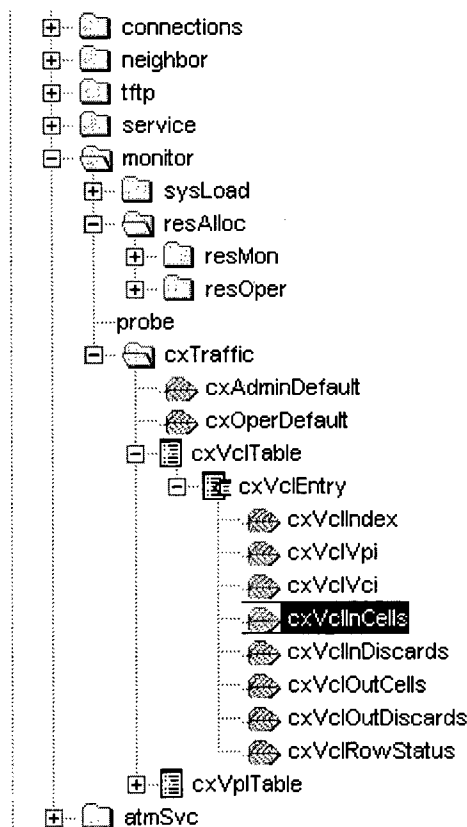
Embora a tecnologia ATM tenha sofrido um processo intenso de maturação nos últimos anos, a incorporação dos níveis de padronização pelos fabricantes dos equipamentos requer uma performance muito maior nos processadores, para acompanhar os protocolos ATM [45], que possuem altas taxas de chamadas de configuração de conexões, e baixas taxas de latência. Além disso, muitas funções precisam ser implementadas em Hardware, devido as velocidades extremamente elevadas dos sistemas ATM.

Deve-se ainda considerar que a maioria das especificações do ATM Forum e do IETF foram completadas nos últimos dois ou três anos, como são os casos da sinalização ILMI, gerência de tráfego, PNNI, LANE, MPOA e gerência de rede.

Como resultado, poucos comutadores da primeira geração (se houverem) têm a capacidade física necessária para implementar essas especificações, que são extremamente complexas e demandam recurso caros do equipamento [45]. De fato, o setor da tecnologia voltado ao ATM está enfrentando problemas que são comuns em outras áreas. Os fabricantes pioneiros começam trabalhando com pré-padrões ou soluções proprietárias. Conforme a indústria do ATM ganhe maturidade, esses mecanismos proprietários devem ser suplantados pelos padronizados.

Devido a essa carência, não é possível, atualmente, realizar gerência de tráfego pelos padrões do ATM Forum. Algumas variáveis auxiliares (extensões de MIBs padronizadas e MIBs proprietárias) foram implementadas pelos fabricantes dos equipamentos do ambiente em estudo (IBM, 3Com e Cabletron). Serão abordadas aqui as variáveis de interesse para o controle de tráfego presentes na IBM ATM-SWITCHING-NODE-MIB v. 4.1.12.

Uma seção da árvore da MIB proprietária da IBM pode ser vista na Figura 7-4.



**Figura 7-4 - Visão parcial da estrutura da MIB IBM ATM-SWITCHING-NODE-MIB v. 4.1.12**

## **Grupo CXTraffic**

Neste grupo, obtém-se uma lista dos canais virtuais para os quais são ativados contadores, um por conexão. Este grupo de contadores é ativado configurando-se duas variáveis, *cxAdminDefault* (retorna o modo desejado para os contadores) e *cxOperDefault* (retorna o valor atual do estado dos contadores). Seus valores podem ser iguais a 1, significando que nenhum contador foi ativado, ou igual a 2, quando deseja-se monitorar “todas” as conexões. O termo está com grafia diferente porque o número de conexões que podem ser monitoradas é limitado, segundo a especificação do fabricante. A mesma especificação não determina qual é este limite, mas observa-se que algumas portas físicas deixam de ser monitoradas. Não se conseguiu estabelecer o critério de não monitoração de determinadas portas.

Objeto *cxVcInCells* : Determina as células recebidas em uma conexão.

Objeto *cxVcInDiscards* : Determina as células descartadas em uma conexão desde que o monitoramento foi inicializado.

Objeto *cxVcOutCells*: O número de células transmitidas pela conexão desde o início do monitoramento.

Objeto *cxVcOutDiscards* Determina as células que seriam transmitidas, mas foram descartadas

## **Grupo SvcTable**

Objeto *atmSvcInterfaceIndex* O valor do *ifIndex* da interface usada pela SVC.

Objeto *atmSvcSiVpi* : O valor do *Vpi* que, em conjunto com o valor de *Vci* retornado por *atmSvcSiVci*, define o canal de sinalização para essa conexão. Usualmente, existe apenas um canal por interface, definido por *VPI=0*, *VCI=5*.

Objeto *AtmSvcSiVci*: Possui o mesmo significado que a variável anterior, apenas representando o identificador do canal.

Objeto *AtmSvcCallReference*: O valor de referência da chamada pela especificação Q2931 do ITU-T [33], utilizada por esta SVC.



Objeto *AtmSvcEndPointReference*: Um dos pontos finais referenciados na especificação Q2931 usados por esta SVC. Se a conexão SVC for *Unicast*, existe apenas uma entrada na tabela. Se for *Multicast* existirá uma entrada para cada parceiro da conexão.

Objeto *AtmSvcCallingNumberAtmAddress*: O número que efetuou a chamada para a conexão. Essa variável retorna o endereço ATM da interface da fonte.

Objeto *AtmSvcCalledNumberAtmAddress*: A mesma definição da variável anterior, apenas contendo o endereço ATM da interface de destino.

Objeto *atmSvcClear*: Esta variável permite ao gerente da rede cancelar esta conexão. Quando a conexão for cancelada (tanto através de uma ação de gerência como por iniciativa da fonte que originou a chamada), a entrada é apagada da tabela, e outra entrada é criada na *atmSvcClearTable*.

Objeto *atmSvcCreationTime*: Dia e hora no qual a conexão foi estabelecida. O retorno é em hexadecimal, e possui o formato apresentado na **Tabela 7-2**.

Octeto	Conteúdo	Intervalo
1-2	Ano	0..65535
3	Mês	1-12
4	Dia	1-31
5	Hora	0-23
6	Minutos	0-59
7	Segundos	0-60 <sup>11</sup>
8	Décimos de segundo	0-9

**Tabela 7-2 - Formato DateTime (RFC 1443)**

Objeto *atmSvcVpi*: O valor de VPI usado pela conexão

Objeto *atmSvcVci*: O valor de VCI usado pela conexão

<sup>11</sup> Os 61 segundos (0 a 60) devem-se a introdução de um segundo incremental - *leap second*. Praticamente todos os sistemas operacionais modernos assumem que um dia corresponde a  $24 \times 60 \times 60 = 86400$  segundos em todos os casos. Pelo padrão de horário do UTC (Universal Time Coordinate), entretanto, cerca de uma vez por ano ou a cada dois anos existe um segundo extra, o *leap second*. Esse segundo incremental sempre é adicionado no último segundo do dia 31 de dezembro ou 30 de junho.

## Grupo InterfaceEntry

Cada entrada nesta tabela corresponde a uma porta que pertence a um módulo ATM do comutador.

Objeto *interfaceMediaErrors*: O número de erros na camada física da interface, tais como erros de violação ou de tamanho.

Objeto *interfaceAvailableBandwidth*: A largura de banda disponível para conexões com banda reservada. O retorno é em bits por segundo (bps).

Objeto *interfaceAllocatedBandwidth*: A largura de banda utilizada, em bps, pelas conexões correntes que possuem reserva de largura de banda nesta porta.

Objeto *interfaceMaxBandwidth*: para uma porta com link PNNI, significa o máximo de largura de banda utilizável pela porta. Esse valor não pode exceder a largura de banda física da porta, dada por *interfaceMediaSpeed*. O valor mínimo válido para essa variável é 60000 bps, que é a largura de banda requisitada para uma conexão PNNI sem reserva de largura de banda. Essa variável deve ter o mesmo valor nas duas extremidades do link PNNI, para evitar deadlocks nos cálculos de roteamento. O valor padrão é igual ao da variável *interfaceMediaSpeed*. Para portas não PNNI, essa variável não pode ser modificada, e permanece sempre igual a *interfaceMediaSpeed*. Esse valor somente pode ser modificado quando o estado da porta estive *disabled*.

Objeto *interfaceFrameFormat*: Retorna o formato dos quadros trocados nesta porta. Para uma porta LAN 155 Mbps, dois formatos são suportados: *sonet-sts-3c* e *sdh-stm-1*. A variável só pode ser modificada se o acesso a porta for UNI ou NNI. Para os demais acessos, o formato do quadro não pode ser modificado e sempre retorna sempre "none". O formato do quadro somente pode ser modificado quando o estado da porta estiver *disabled*.

Os valores retornados estão listados na Tabela 7-3.

Formato do Quadro	Valor de retorno
none	1
sonet-sts-3c	2
sdh-stm-1	3

---

ds3	4
e3	5
e1	6
t1	7
sonet-sts-12c	8

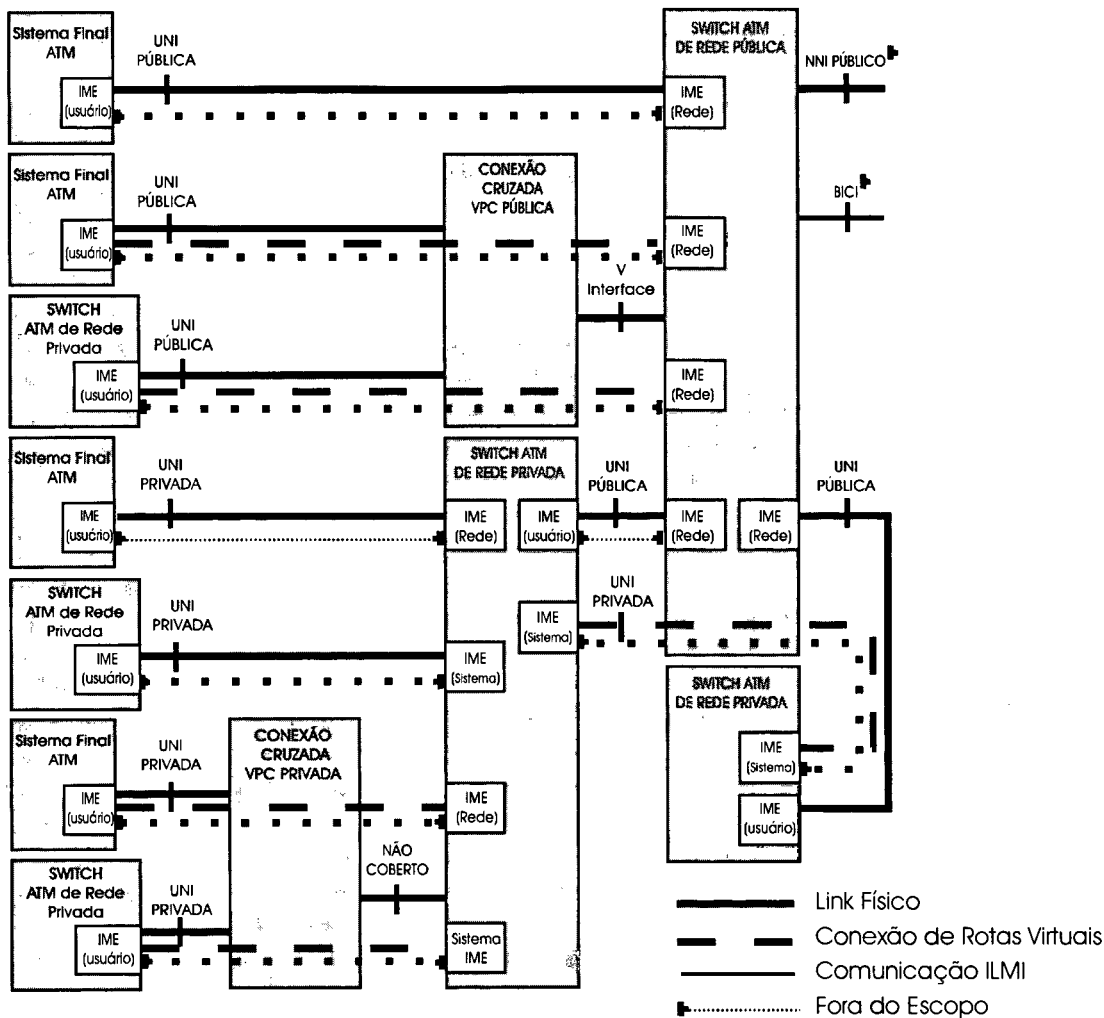
**Tabela 7-3 - Valores retornados pela variável *interfaceFrameFormat***

Outros objetos foram estudados e selecionados, mas não estão sendo utilizados na versão atual da ATM . Tais objetos estão detalhados no Anexo I

## 7.4.ILMI

A *Integrated Local Management Interface (ILMI)* foi definida pelo ATM Forum para assegurar a interoperabilidade entre comutadores de diferentes fabricantes. Quando foi especificada originalmente, esperava-se que fosse um padrão provisório. O nome inicial da especificação era “Interim Local Management Interface”, até que o ITU-T completasse sua especificação de gerência.

O propósito da ILMI é possibilitar que dois dispositivos ATM adjacentes façam a configuração automática dos parâmetros operacionais referentes ao link que compartilham. Além disso, devem trocar entre si informações de gerência. Em particular, essa interface é usada para fornecer para os dispositivos ATM informações sobre o status e a configuração da interface da camada física, da interface da camada ATM e dos VPCs e VCCs do sistema adjacente. A ILMI foi definida inicialmente para as UNI pública e privada. Entretanto, a última versão, ILMI 4.0, foi estendida para suportar PNNI. A ILMI situa-se no modelo geral de gerência para um dispositivo ATM conforme a **Figura 7-5**. Em adição, a especificação ILMI possibilita aos usuários e operadores da rede a utilização de funções de gerência que atualmente não estão disponíveis a partir de outras tecnologias de uma forma padronizada e consistente. Essa capacidades incluem comunicação bi-direcional através de um link ATM possibilitando a ambos os lados verificar parâmetros de contrato, e a configuração dos novos serviços de tráfego como o ABR, identificação da versão de sinalização e informações sobre o registro de endereços.



**Figura 7-5 -ILMI versão 4.0**

Uma Entidade de Gerência de Interface (IME – Interface Management Entity) é associada com cada interface ATM que suporte as funções da ILMI para aquela interface. Existe um conjunto de objetos gerenciáveis para cada interface ATM, representando os atributos ILMI, suficientes para suportar as funções ILMI em cada interface ATM. Os atributos ILMI por interface ATM são organizados em uma estrutura de MIB padrão; existe uma instância da estrutura da MIB para cada interface ATM em cada dispositivo ATM. Em consequência, um dispositivo ATM pode ter múltiplos IMEs, se suportar múltiplas interfaces ATM.

## O protocolo ILMI

As comunicações do ILMI ocorrem entre dois IMEs adjacentes, sobre links físicos ou virtuais, como os VPCs usados pelo protocolo IISP (*Interim Interswitch*

*Signaling Protocol*) ou PNNI (Private Network-Network Interface). Para não definir um novo protocolo, o ILMI foi projetado para usar os mesmos formatos de mensagens e a semântica do SNMPv1. Assim, o ILMI possui os mesmos cinco tipos de PDU do SNMPv1 (get, get-next, set, get-response e trap) e as variáveis incluídas nessas PDUs são definidas em uma MIB usando a SMI do SNMP v1.

O protocolo de comunicações ILMI é um protocolo aberto (como o SNMP e a AAL) Figura 7-6. Uma IME pode acessar, através da ILMI, a informação da MIB ILMI IME associada com sua IME adjacente. Um VCC pré-definido (*Well-Know*) é usado pelo ILMI. O valor padrão para essa conexão é VPI=0, VCI=16, mas os valores de VPI/VCI são configuráveis.

Entretanto, o ILMI tem um paradigma diferente do usado pelo SNMP para a gerência de redes. Na gerência de redes, um gerente dispara uma requisição e um agente responde com uma *PDU response* ou emite *traps*. Em contraste, o ILMI é usado para a gerencia de interfaces entre duas IMEs, um em cada lado da interface. Ambas IMEs podem enviar *requests*, bem como *responses* ou *traps*. As duas IMEs possuem seus próprios valores (potencialmente diferentes) para os mesmos objetos da MIB.

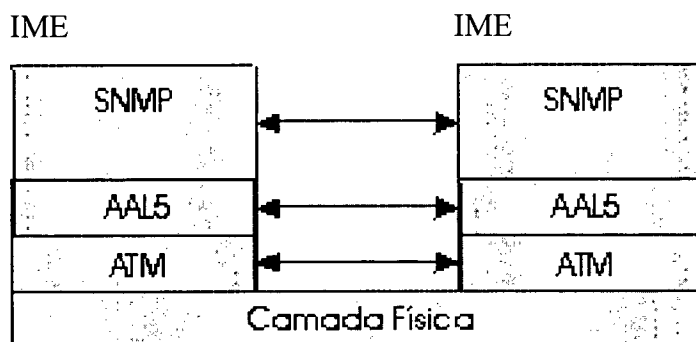


Figura 7-6 - Comunicação entre duas IMEs através da ILMI

## ILMI MIB

A MIB ILMI consiste de 4 módulos, demonstrados na Tabela 7-4.

Módulo	Conteúdo
Módulo de convenções textuais	Define um número de convenções textuais comuns, e OIDs, de forma que outros módulos de MIBs possam importá-lo de forma mútua e consistente.
MIB de gerência do Link	Fornece facilidades de gerência do link ATM de propósito geral.
MIB de registro do endereço	Informações para registro de endereços
MIB de registro de serviço	Extende a MIB ATM para fornecer registros de serviços de propósito geral para localizar serviços da rede ATM como o LECS.

Tabela 7-4 - Módulos da MIB ILMI

### VPC/VCC Configuration Information

Os objetos da categoria configuração dos VPC / VCC contêm as informações pertinentes à capacidade de comutação do comutador ATM. Deve-se notar que, apesar de existir a divisão dos mecanismos de comutação em comutadores VPC / VCC e cross-connects, não significa que o comutador ATM tenha dois tipos físicos de mecanismos de comutação (*switch fabrics*). Tipicamente apenas uma fabrica é usada para as conexões permanentes e comutadas. Entretanto, os comutadores ATM usualmente reservam diferentes faixas de números de VPC / VCC para as conexões permanentes e as comutadas. Além disso, certos comutadores podem suportar somente uma certa faixa de números de VPI / VCI devido às limitações das suas próprias implementações, ou seja, a capacidade da matriz de comutação. Para assegurar a interoperabilidade, o ILMI propicia um mecanismo para os comutadores ATM nos dois extremos das interfaces para negociar um conjunto de parâmetros mutuamente suportados.

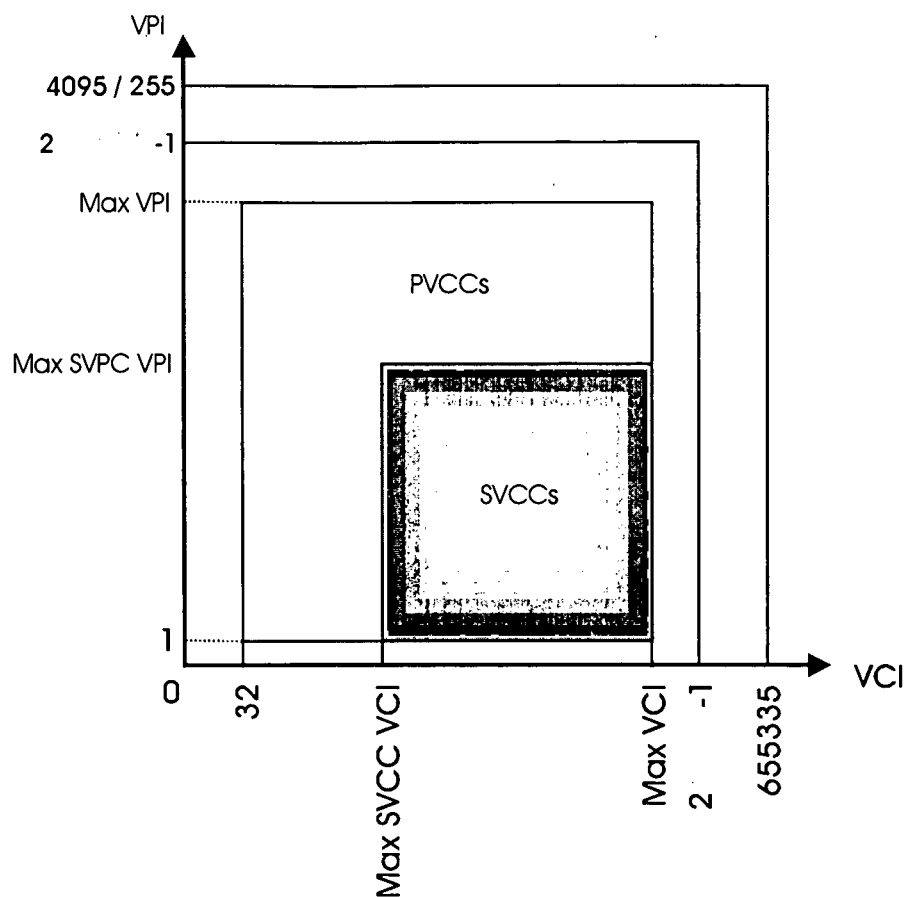


Figura 7-7 - Número de VPIs e VCIs

A alocação dos VPI / VCI está demonstrada na Figura 7-7. Segundo Pan [45], os VPCs comutados para uma interface ATM começam de uma única faixa contígua de VPIs, a partir do VPI=1 (VPI=0 é usado para o ILMI e a sinalização dos VCCs) e terminando com o número indicado pelo objeto “Máximo VPC VPI comutado” (*atmfAtmLayerMaxSvpcVpi*). Neste estudo, todas as conexões comutadas apresentaram seu VPI=0, independente de serem conexões de sinalização e controle ou requisitadas pelo usuário. Embora Conexões virtuais permanentes (VPC) possam ser alocadas em qualquer local, a ILMI sugere que as conexões virtuais permanentes tenham seus números de Identificadores de rotas virtuais (VPI) maiores que o máximo VPI dos SVPC (conexões de rotas virtuais permanentes) e menores que o máximo VPI.

Novamente, as VCCs (conexões de canais virtuais) comutadas para uma dada interface ATM são alocados de uma faixa contígua de VCIs iniciando com o número indicado pelo menor objeto VCI VCC (*atmfAtmLayerMinSvccVci*) e terminando com o máximo VCI. O mesmo valor se aplica para todos os VPI VCCs para os quais a pilha de sinalização está configurada. Para ser compatível com as especificações do ATM Forum para os circuitos virtuais bem-conhecidos (Well Know), esse valor deve ser, pelo



menos, 32. As conexões de canais virtuais (VCCs) podem ser alocadas em qualquer local, mas a ILMI sugere que os VCCs permanentes tenham valores de VCI menores que os mínimos VCI SVCC.

Devido a limitações de hardware e software, o número máximo de VPC / VCC é menor ou igual a 2 elevado ao número máximo de bits ativos de VPI/VCI. Essa variável pode ser coletada através da MIB ATM MIB 2, pelo grupo *interfaceConfEntry*, objeto *atmInterfaceMaxActiveVciBits*

IMEs adjacentes negociam parâmetros de configuração na inicialização do ILMI. Cada IME recupera os valores dos objetos *MAXIMUM* dos seus pares, incluindo o número máximo de bits VPI/VCI ativos, o número máximo de VPC/VCC e o número máximo de VPI SVPC / VPI SVCC. Os bits VPI/VCI ativos identificam os bits nos campos VPI/VCI que são usados para a comutação de células. O valor recuperado é comparado com o valor local de cada objeto. O valor atual usado é definido para o menor dos dois para garantir a interoperabilidade. Similarmente, cada IME recupera o valor mínimo dos objetos VPI/VCI comutados. O valor atual usado é o maior dos dois.

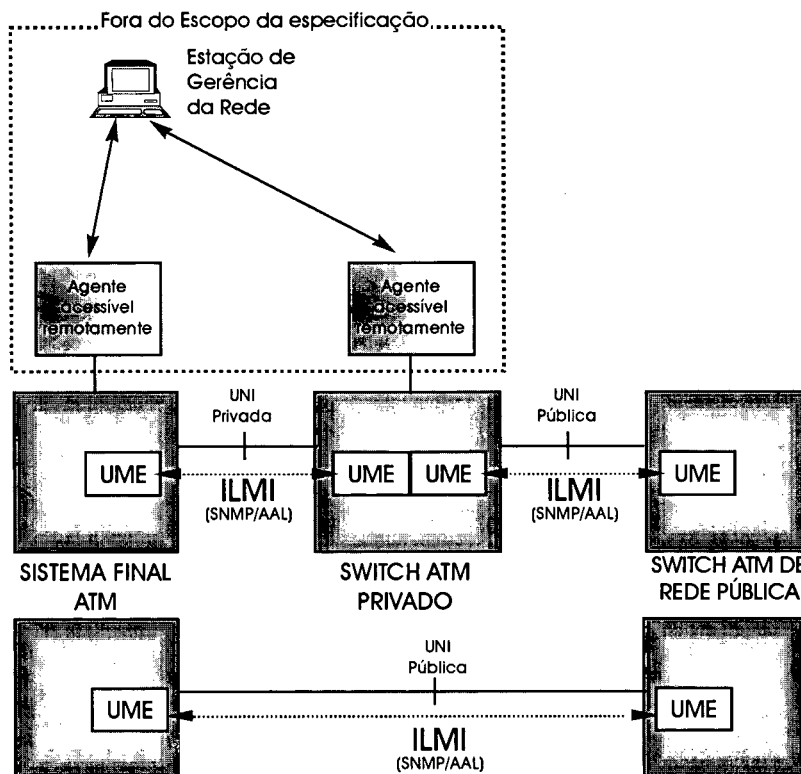
## Informações de Protocolo

Com os objetos de informação de protocolos definidos nessa categoria, o ILMI é capaz de identificar automaticamente o tipo de conexão e protocolos de sinalização suportados pela entidade par. A informação inclui o tipo de interface ATM (pública ou privada) o tipo de dispositivo (user ou network); a última versão dos protocolos ILMI/UNI/PNNI suportados pela interface.

IMEs adjacentes também negociam protocolos adequados, que sejam suportados por ambas interfaces. Se o valor de um objeto em uma versão do protocolo é o mesmo ou posterior que o valor da IME local, então o valor da IME local deve ser acatado. Senão, se o valor da IME par do objeto é mais recente e suportada localmente, a IME local deve acatar o valor da IME remota (peer IME). Esse procedimento garante a compatibilidade dos protocolos usados entre as IMEs adjacentes..

As MIBs da ILMI foram especificadas inicialmente pela UNI do ATM Forum [62]. Logo a seguir, nova especificação deu um novo status para a ILMI [64]. Na UNI 3.1, o ATM Forum declara vários princípios e características para a interface, representados na Figura 7-8. Nessa versão da UNI, o ATM Forum não especificava

nenhuma forma de acesso a MIB da ILMI que não fosse pelas interfaces UNI adjacentes (UMEs – UNI Management Entity), deixando a implementação do acesso ao cargo de cada fabricante. Na especificação da ILMI 4.0, é recomendada a utilização de um agente Proxy, conforme a Figura 7-10



**Figura 7-8 - Definição e contexto da ILMI - UNI 3.1**

Não é possível acessar-se a MIB ILMI através de uma estação de gerência, sem a utilização de um *proxy*.

### Agente Proxy

Em adição ao seu papel para gerência de interfaces locais, os dados das MIBs ILMI também são úteis para funções genéricas de gerência de redes. A ILMI define um mecanismo de agente proxy, que utiliza as funções existentes na ILMI para possibilitar uma NMS acessar a MIB da interface ATM. O proxy utiliza a capacidade de agente do IME para acessar a MIB da interface ATM no sistema vizinho. A solução está representada na **Figura 7-10**

O agente proxy SNMP aceita um *request* SNMP a partir de uma estação de gerência (NMS) e repassa para a IME apropriada para processamento, tanto local (contra sua própria MIB) ou como uma operação a ser mandada adiante, para processamento em sua IME par, remotamente.

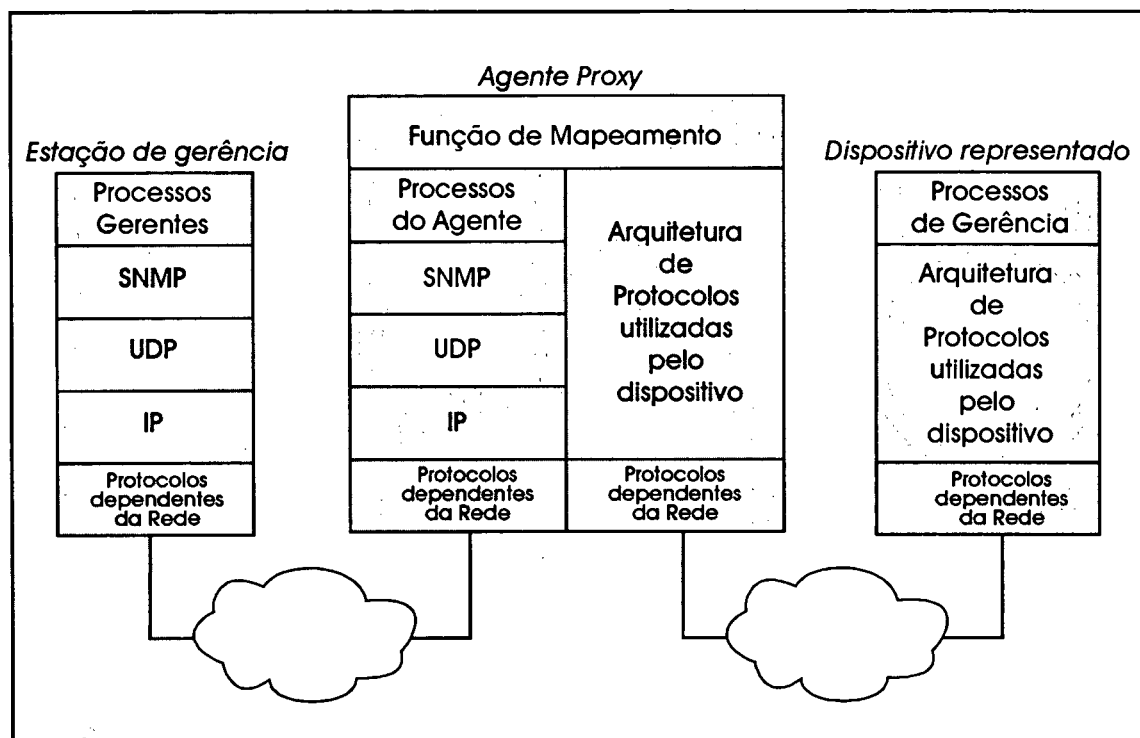
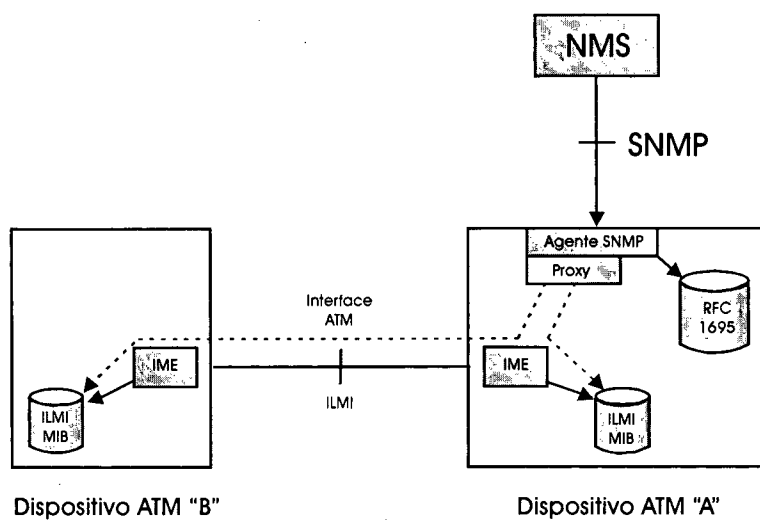


Figura 7-9 - Configuração de proxies SNMP ([55]).

Para cada IME, dois alvos proxy são definidos, um para receber os requests para a interface ATM local e outro para tratar os requests para os dados da interface vizinha.. Quando o agente proxy recebe um *request* de uma estação de gerência (NMS) ele primeiro determina, em termos da string da comunidade no *request* se ele deve tratar o request normalmente ou repassá-lo para um dos agentes proxy alvos. Por exemplo, um request com a string de comunidade "community" identifica a mensagem a ser enviada para a ser enviada para a RFC 1695 como usualmente. Se a string é "local", o request será repassado para a IME local, enquanto o pacote "remoto" será enviado a IME adjacente.



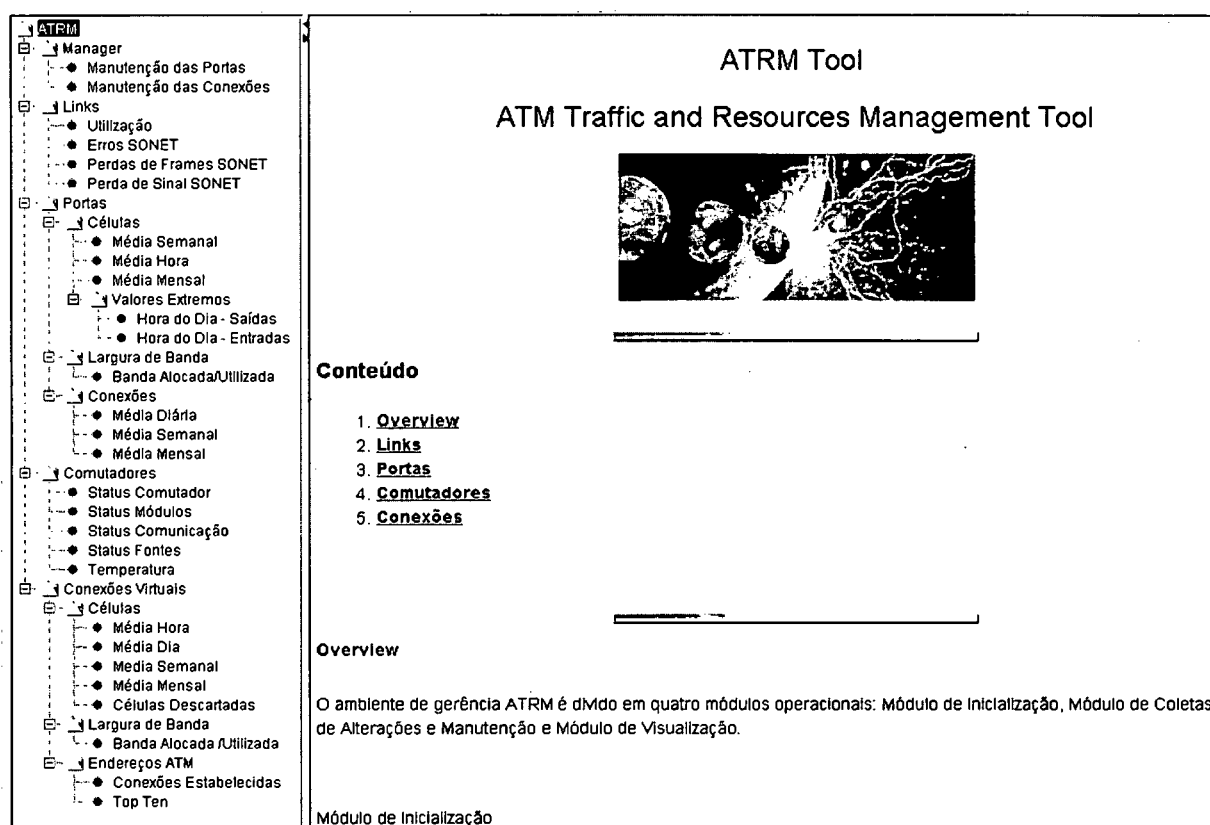
**Figura 7-10 - Acesso NMS a ILMI através de agente proxy**

Quando o destino proxy recebe o request ele efetua a operação SNMP com respeito as suas definições na sua MIB e manda a resposta (*response*) para o agente proxy, que, por sua vez, retorna a informação para a NMS.

Devido a imposição da utilização de um *proxy*, as MIBs da ILMI não serão utilizadas no ambiente de gerência ATM.

## 8. O Ambiente de gerência ATRM-WTool

O ambiente de gerência ATRM-WTool (**Figura 8-1**) é dividido em quatro módulos operacionais: Módulo de Inicialização, Módulo de Coletas, Módulo de Alterações e Manutenção e Módulo de Visualização (**Tabela 8-2**). Esses módulos serão descritos a seguir, quanto sua funcionalidade, modo de operação e facilidades apresentadas ao gerente da rede.



**Figura 8-1 - Tela de Abertura do ATRM-WTool**

### 8.1. Módulo de Inicialização

#### 8.1.1. Função

A função do módulo de inicialização é montar os vetores necessários para armazenar as variáveis nas tabelas do Banco de Dados.

### 8.1.2. Modo de Operação

As classes pertencentes a esse módulo do software percorrem as MIBs dos agentes no equipamento indicado pelo gerente da rede. As variáveis buscadas dizem respeito ao número de interfaces, quais estão ativas, quais pertencem ao grupo de interfaces do ifType 37, que corresponde as interfaces ATM (seção 7.1). De posse dessas informações, o módulo passa dimensionar as matrizes necessárias ao armazenamento de cada objeto escolhido pelos outros módulos do sistema.

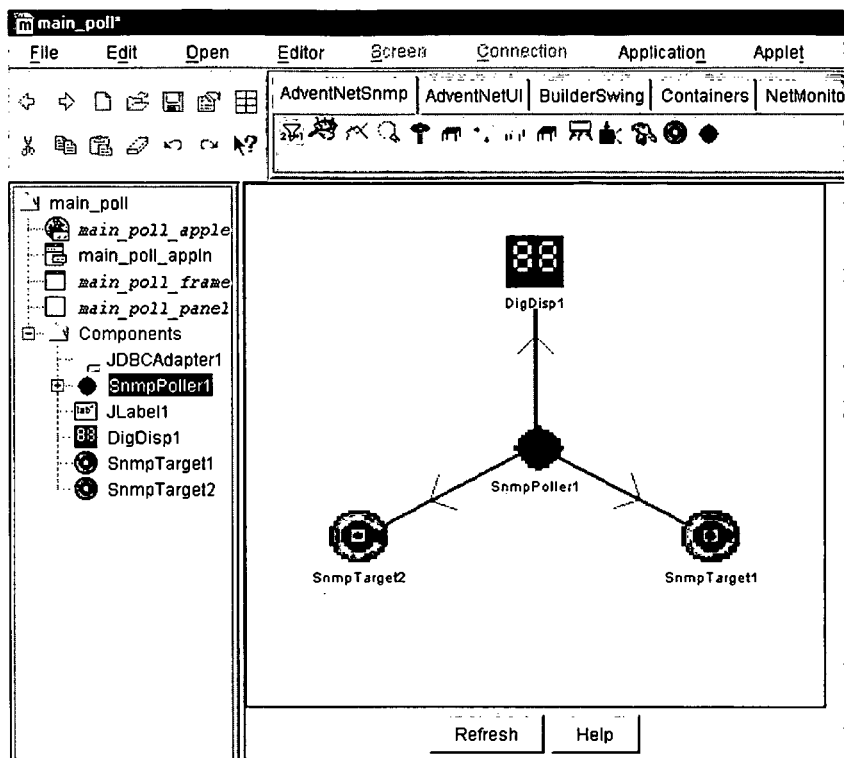
As informações necessárias para o funcionamento do módulo de inicialização são as seguintes:

- Número IP dos comutadores a serem monitorados
- Nome das comunidades de *read*, *write* e *trap* de cada comutador.

De posse dessas informações, as classes Java das bibliotecas SNMP percorrem os endereços fornecidos, verificando os parâmetros descritos na seção anterior. As classes envolvidas são as seguintes:

- SNMPPoller

A classe `SnmpPoller` é a responsável por disparar o evento de coleta das informações. Um objeto instanciado dessa classe, denominado `SNMPPoller1`, possui as informações pertinentes ao intervalo de *polling* (envio de PDUS SNMP Request), direcionados para dois objetos alvo, instanciados da classe `SNMPTarget`. A **Figura 8-2** mostra o relacionamento entre esses objetos. Para o ambiente de desenvolvimento Adventnet Builder, para cada ligação entre dois objetos é gerada uma nova classe de conexão entre ambos, responsável pelos eventos no objeto de destino. Nessa classe de conexão, o objeto de destino, instanciado como “target”, recebe parâmetros para seus métodos. No exemplo da **Figura 8-2**, os “alvos” das conexões que partem do objeto `SNMPPoller1` são os objetos `DigDisp1`, `SNMPTarget1` e `SNMPTarget2`. Como exemplo de passagem de parâmetros, podemos citar a instrução [ *target.setNumericValue(arg0.getNumericValue()/100/60/60 );*]. Tal instrução, passada pela conexão do `SNMPPoller1` para o `DigDisp1`, determina que o display numérico da tela assumo o valor coletado pela variável SNMP configurada no objeto `SNMPPoller1`. A divisão por (100/60/60) faz com que o valor seja mostrado na tela em unidades de horas, uma vez que o retorno da variável coletada é em milissegundos.



**Figura 8-2 - Relacionamento entre as classes na ferramenta de desenvolvimento Adventnet Builder**

- SNMPTarget

A classe SNMPTarget possui duas instâncias, SNMPTarget1 e SNMPTarget2. Esses objetos são responsáveis pelo armazenamento das variáveis SNMP, em forma de um vetor unidimensional, como mostrado na **Figura 8-3** e na **Tabela 8-1**. No momento programado pelo objeto SNMPPoller, inicia-se a coleta dos valores contidos no vetor.

- JDBCAdapter

A classe JDBCAdapter é responsável pela conexão com o banco de dados, como descrito na seção 8.2.2.

- JLabel

Essa classe é utilizada apenas para formatar texto na interface gráfica da aplicação. Pode-se configurar os parâmetros relativos a posição na tela, tipo, cor e tamanho da fonte, cor de fundo.

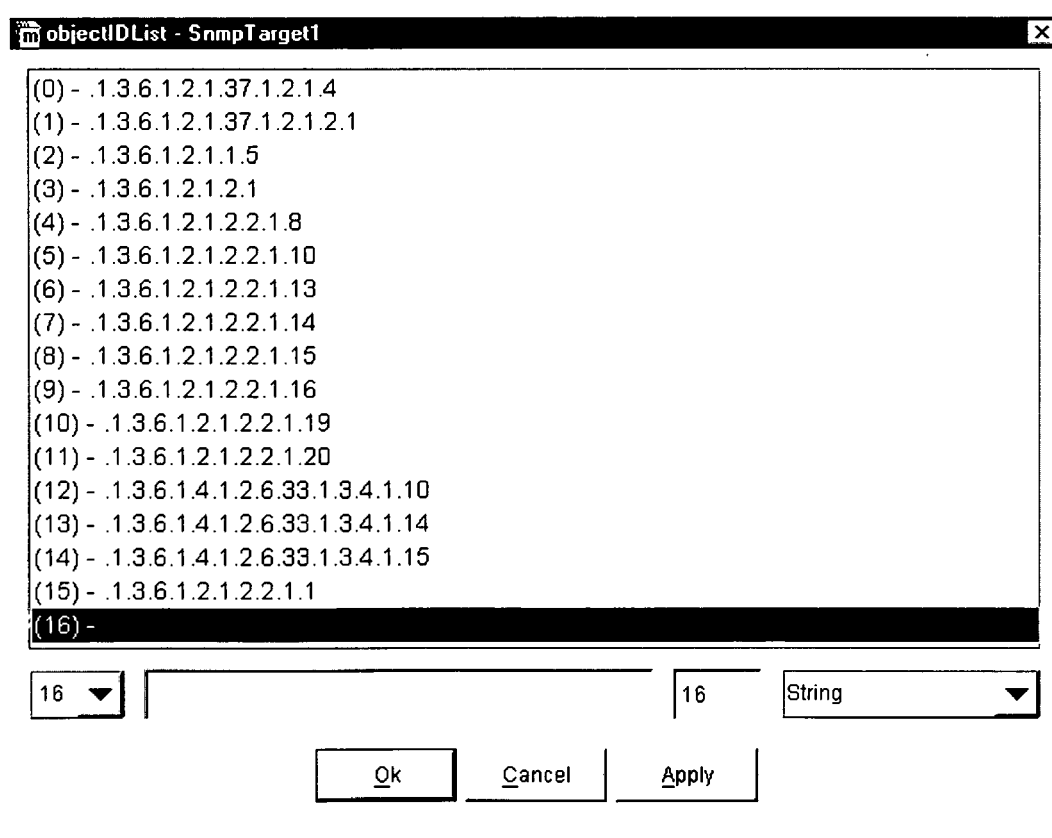
- DigDisp

A classe DigDisp é utilizada para instanciar objetos de display numérico na interface gráfica. No módulo de inicialização, um desses objetos é utilizado para

informar ao gerente o número de horas decorridas desde o último *reset* da estação de gerência (NMS).

MIB	Índice array	OID	Nome
ATOM	0	.1.3.6.1.2.1.37.1.2.1.4	AtmInterfaceConfVccs
ATOM	1	.1.3.6.1.2.1.37.1.2.1.2.1	AtmInterfaceMaxVccs
MIB 2	2	.1.3.6.1.2.1.1.5.0	SysName
MIB 2	3	.1.3.6.1.2.1.2.1.0	ifNumber
MIB 2	4	.1.3.6.1.2.1.2.2.1.8	IfOperStatus
MIB 2	5	.1.3.6.1.2.1.2.2.1.10	IfInOctets
MIB 2	6	.1.3.6.1.2.1.2.2.1.13	IfInDiscards
MIB 2	7	.1.3.6.1.2.1.2.2.1.14	IfInErrors
MIB 2	8	.1.3.6.1.2.1.2.2.1.15	IfInUnknowProtos
MIB 2	9	.1.3.6.1.2.1.2.2.1.16	ifOutOctets
MIB 2	10	.1.3.6.1.2.1.2.2.1.19	ifOutDiscards
MIB 2	11	.1.3.6.1.2.1.2.2.1.20	ifOutErrors
IBM	12	.1.3.6.1.4.1.2.6.33.1.3.4.1.10	InterfaceMediaErrors
IBM	13	.1.3.6.1.4.1.2.6.33.1.3.4.1.14	interfaceAvailableBandwidth
IBM	14	.1.3.6.1.4.1.2.6.33.1.3.4.1.15	InterfaceAllocatedBandwidth
MIB 2	15	.1.3.6.1.2.1.2.2.1.1	ifIndex
MIB 2	16	.1.3.6.1.2.1.2.2.1.7	ifAdminStatus

**Tabela 8-1 - Descrição das variáveis SNMP, MIBs e posição no vetor de inicialização.**



**Figura 8-3 - Vetor de variáveis SNMP para objeto SNMPTarget1, elaborado no ambiente de desenvolvimento.**



### 8.1.3. Facilidades apresentadas ao gerente

O módulo de inicialização permite uma coleta dinâmica das variáveis, sem que o gerente se preocupe em ajustar quantas portas estão em operação em cada comutador, e quais delas correspondem a interfaces ATM. Dessa forma, quando é acrescentado um novo slot ao chassi modular dos comutadores, as variáveis dessas novas portas são automaticamente acrescentadas no vetor de coletas, sendo então armazenadas nas tabelas do banco de dados.

## 8.2. Módulo de Coletas

### 8.2.1. Função

Utiliza os objetos selecionados das MIBs descritas na seção 7, e através de primitivas *SNMP Request*, armazena os resultados nas tabelas do banco de dados.

O módulo de coletas realiza outra tarefa básica, de manutenção das tabelas do banco de dados, calculando as médias horárias, diárias, semanais, mensais e anuais das tabelas básicas, e armazenando os resultados em tabelas menores, o que permite uma racionalização da utilização do espaço em disco.

### 8.2.2. Modo de Operação

A inserção nas tabelas é realizada utilizando-se a classe *java.sql* do JDK 1.2. Essa classe possui duas formas de conexão aos bancos de dados: Através do ODBC ou através de “*Open-Client DB-Library*”, um conjunto de bibliotecas que acessam as APIs dos bancos de dados, comumente denominados “*drivers*”. Optou-se pela utilização desse último tipo de conexão, uma vez que não necessitam utilizar o ODBC do cliente da aplicação, o que gera restrições de segurança nos *applets*. O processo utilizado pelas aplicações que utilizam uma biblioteca “*Open-Client DB-Library*” é o seguinte [69]:

- Estabelecer uma conexão com o servidor SQL;
- Montar um conjunto de instruções (*SQL Statements*) em um *buffer*;
- Submeter o conjunto de instruções ao servidor SQL;
- Seqüencialmente, processar as instruções e recuperar os resultados;
- Fechar a conexão.

Tais bibliotecas clientes normalmente não estão disponíveis como “domínio público”, e precisam ser adquiridas dos fornecedores. Neste estudo, optou-se por utilizar uma biblioteca de versão experimental (*trial version*) fornecida pela I-net Software (MS

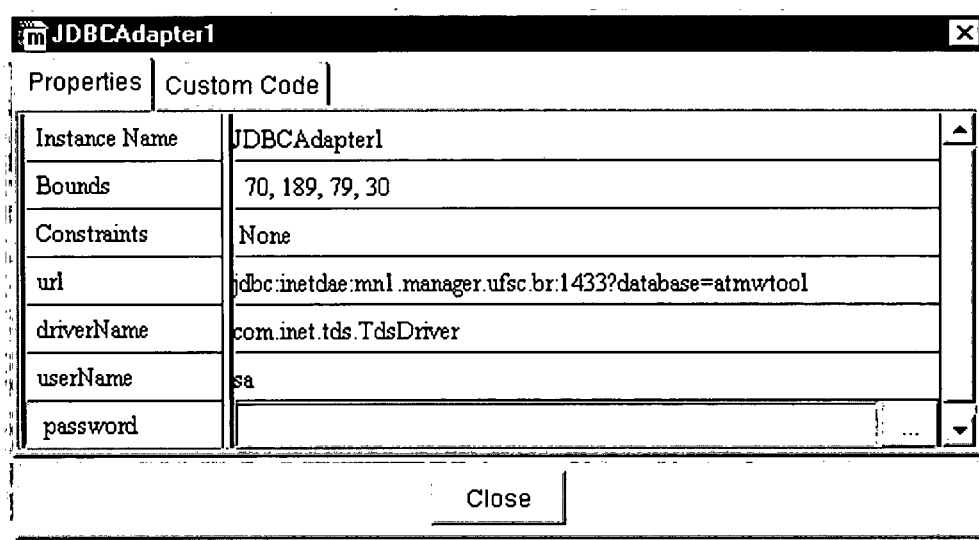
SQL JDBC Driver Version 1.18) que permite no máximo duas conexões simultâneas ao banco de dados. Tal restrição não causa problemas maiores em um ambiente de estudos, onde a aplicação de gerência é acessada por poucos usuários. Essa biblioteca pode ser encontrada na Web em <http://www.inetsoftware.de>.

As requisições do módulo de coleta são efetuadas a cada 300 segundos, e armazenadas nas tabelas do DBMS. A estrutura da tabela utilizada para coleta das conexões virtuais e fluxo de células nas portas físicas está representada na **Figura 8-11**.

As classes envolvidas nas telas do módulo de coletas são as seguintes:

- JDBCAdapter

Esta classe está presente no pacote de classes do AdventNet Builder, e possui a função básica de conectar a aplicação a um banco de dados, através da utilização de APIs específicas para cada tipo de DBMS. As propriedades iniciais da classe são inicializadas através da interface gráfica do AdventNet Builder, como pode ser observado na **Figura 8-4**.



**Figura 8-4 - Configuração das variáveis para classe JDBCAdapter**

- SNMPPoller e SNMPTarget

O modo de operação destas duas classes já foi descrito na seção anterior, 8.1.2.

### 8.2.3. Facilidades apresentadas ao gerente

O módulo de coletas apresenta a vantagem de armazenar, de forma transparente, todas as variáveis SNMP contempladas na seleção efetuada na seção 7, que tratou da análise das MIBs relacionadas aos recursos e tráfego ATM. Além disso, periodicamente

é disparada uma rotina para armazenar os dados com valores médios, reduzindo assim o tamanho das tabelas. São calculadas e armazenadas automaticamente médias horárias, diárias, semanais e mensais em tabelas auxiliares.

As vantagens de obterem-se dados históricos do comportamento da rede podem ser observadas na seção 9.2.5, onde são discutidos alguns levantamentos estatísticos qualitativos, de forma a se observar as flutuações temporais das variáveis de tráfego e recursos. Com o estabelecimento de um padrão comportamental da rede, podemos criar um “compromisso” gerencial, indicando as diferenças entre a normalidade e a anormalidade dos parâmetros.

### ***8.3. Módulo de Alterações/Manutenção (Manager)***

#### **8.3.1. Função**

Esse módulo permite ao gerente da rede efetuar a manutenção, tanto das tabelas quanto dos objetos gerenciáveis SNMP. O gerente emitir primitivas SNMP SET request, para alterar parâmetros dos comutadores, como:

- Quantidades de canais virtuais comutados permitidos em cada módulo e/ou porta do comutador;
- Alteração de estados administrativos das conexões, dos módulos ou das portas dos comutadores;
- Reinicializar os comutadores, os módulos dos comutadores, as portas de cada módulo ou somente os contadores dos agentes SNMP.

Este módulo, através da tela de entrada mostrada na **Figura 8-5** permite ainda efetuar consultas (*SQL select*) genéricas nas tabelas, onde o administrador pode emitir comandos de inserção (*SQL insert*), atualização (*SQL update*). O administrador pode ainda remover dados (*SQL delete*).

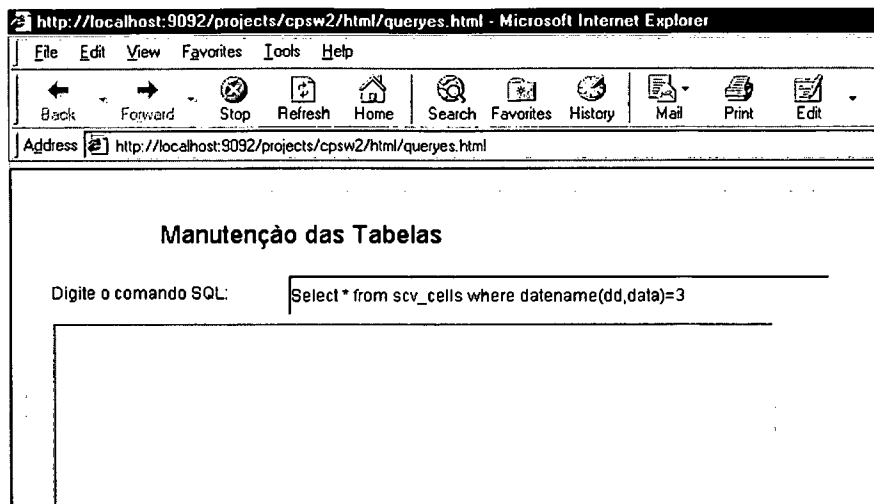


Figura 8-5 - Tela de comandos SQL do módulo manutenção

### 8.3.2. Modo de Operação

O módulo de alterações e manutenção (manager) funciona através de comandos diretos do gerente para a base de dados, através da classe JDBCAdapter (**Figura 8-5**) ou então do envio de primitivas SNMP *set request*, com o auxílio da classe SNMPTarget. Tais primitivas tem o poder de alterar algumas variáveis nas MIBs dos agentes gerenciados, como é o caso de reduzir o número de conexões virtuais possíveis em cada porta ou colocar uma conexão ou porta física em status administrativo “Down” ou “Up”, o que torna esse dispositivo inoperante ou preparado para transmitir células ATM.

A manutenção das tabelas pode ser feita através de comandos na linguagem SQL padrão.

### 8.3.3. Facilidades apresentadas ao gerente

O módulo de alteração/manutenção permite que o gerente da rede ATM mantenha um controle sobre os recursos utilizados, como o número e duração das conexões, bem como uma manutenção sobre os dados históricos da utilização desses recursos. A utilização da linguagem SQL permite uma versatilidade muito grande aos comandos efetuados.

## 8.4. Módulo de Visualização

### 8.4.1. Função

O módulo de visualização permite que o usuário e/ou administrador verifique as situações dos parâmetros relativos ao tráfego e recursos da rede ATM, agrupados em 4 classes [2]: Eventos relativos aos Links, Portas, Comutadores e Conexões (**Figura 8-1**, **Tabela 8-2**). Tais eventos podem ser visualizados em tempo real, quando os parâmetros assim exigirem (alocação de banda, utilização de CPU), ou então pela recuperação das tabelas do banco de dados (valores médios de entrada e saída nas portas, número de conexões ativas)

### 8.4.2. Modo de Operação

O módulo de visualização do ambiente ATM funciona através de classes dedicadas a consultar a base de dados histórica para levantamentos do comportamento da rede ao longo de um período específico, ou então consultar os agentes SNMP dos comutadores de forma “*on line*”, como a monitoração do estabelecimento de conexões em determinadas portas, a temperatura interna dos comutadores e a utilização das CPUs.

Na primeira situação (consulta às tabelas), são utilizadas as seguintes classes:

- JDBCAdapter

Descrita no módulo anterior, seção 8.2.2, essa classe tem a função de estabelecer uma conexão com a base de dados.

- LineGraph, BarGraph

São classes utilizadas para demonstrar os resultados das consultas às tabelas da base de dados, depois de efetuadas as consultas pela JDBCAdapter. O retorno das consultas pode ser visualizado na interface do browser do cliente, como mostrado na **Figura 8-9**.

- JTextField

Utilizada para receber entradas do usuário e relacioná-las com os parâmetros de consulta, como o endereço ATM ou IP do comutador, a conexão virtual ou a porta física a ser consultada.

- SnmpPoller, descrita no módulo de coletas, que dispara primitivas SNMP *Get Request* para os agentes.

### 8.4.3. Facilidades apresentadas ao gerente

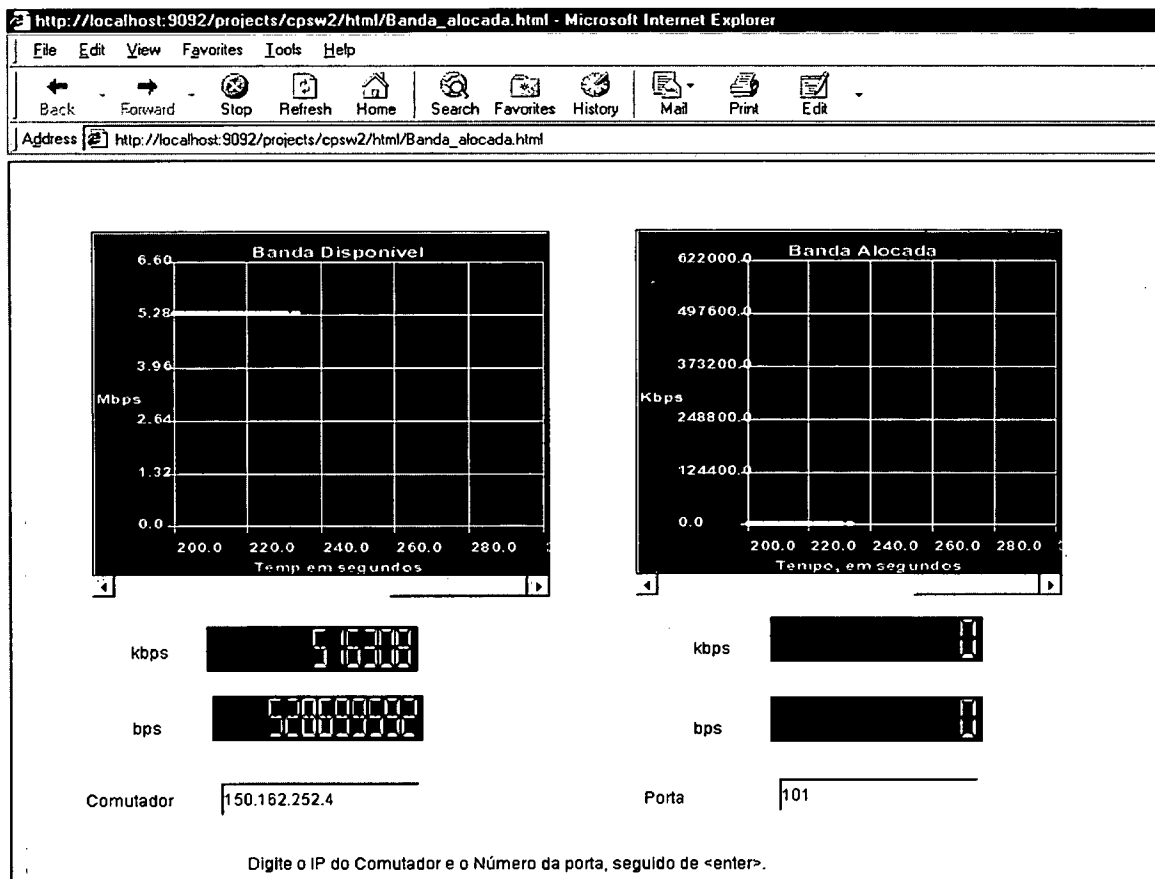
O módulo de visualização permite ao gerente e ao usuário final um monitoramento da utilização da rede, bem como do comportamento histórico dos parâmetros estudados. Uma das vantagens mais salientes do módulo de visualização (e do ambiente ATM) é a possibilidade de observar-se esse comportamento a partir de qualquer ponto da rede, sem a necessidade de se estar na estação de gerência.

<b>Módulo de Inicialização</b>			
<b>Tabelas de Coletas</b>		<b>Tabelas de Médias</b>	
<b>Objetivo</b>	<b>Classes</b>	<b>Objetivo</b>	<b>Classes</b>
Armazenas as coletas realizadas pelos SNMP Requests a cada 300 segundos	Main_pool.class	Calcula as médias da tabela de coletas, agrupando os resultados em uma tabela menor	Tab_media.class
<b>Módulo de Coleta</b>		<b>Módulo de Alterações/Manutenção</b>	
<b>Objetivo</b>	<b>Classes</b>	<b>Objetivo</b>	<b>Classes</b>
Coleta de eventos nas portas	Main_pool.class	Limpar/Visualizar tabela Portas	Core_portas.class
Coleta de eventos nas conexões	Main_pool.class	Limpar/Visualizar Tabela SVC	SVC_criacao.class
<b>Módulo de Visualização</b>			
<b>Eventos no nível dos Links</b>		<b>Eventos no nível de portas</b>	
<b>Objetivo</b>	<b>Classes</b>	<b>Objetivo</b>	<b>Classes</b>
Utilização do link	Link.class	Número de células transmitidas/recebidas	Dia_sem.class Extremos.class Extremos_saidas.class Medhora.class Media_portas.class Media_mes.class Media_ano.class
Erros de paridade no nível SONET	Erros_sonet.class	Banda alocada	Banda_alocada.class
Perda de frames no nível SONET	Perda_frames.class	Banda utilizada	Banda_alocada.class
Perda de Sinal no nível SONET	Perda_sinal.class	Número de Conexões Virtuais Comutadas	Media_conex.class
<b>Eventos no nível de comutadores</b>		<b>Eventos no nível de conexões Virtuais</b>	
<b>Objetivo</b>	<b>Classes</b>	<b>Objetivo</b>	<b>Classes</b>

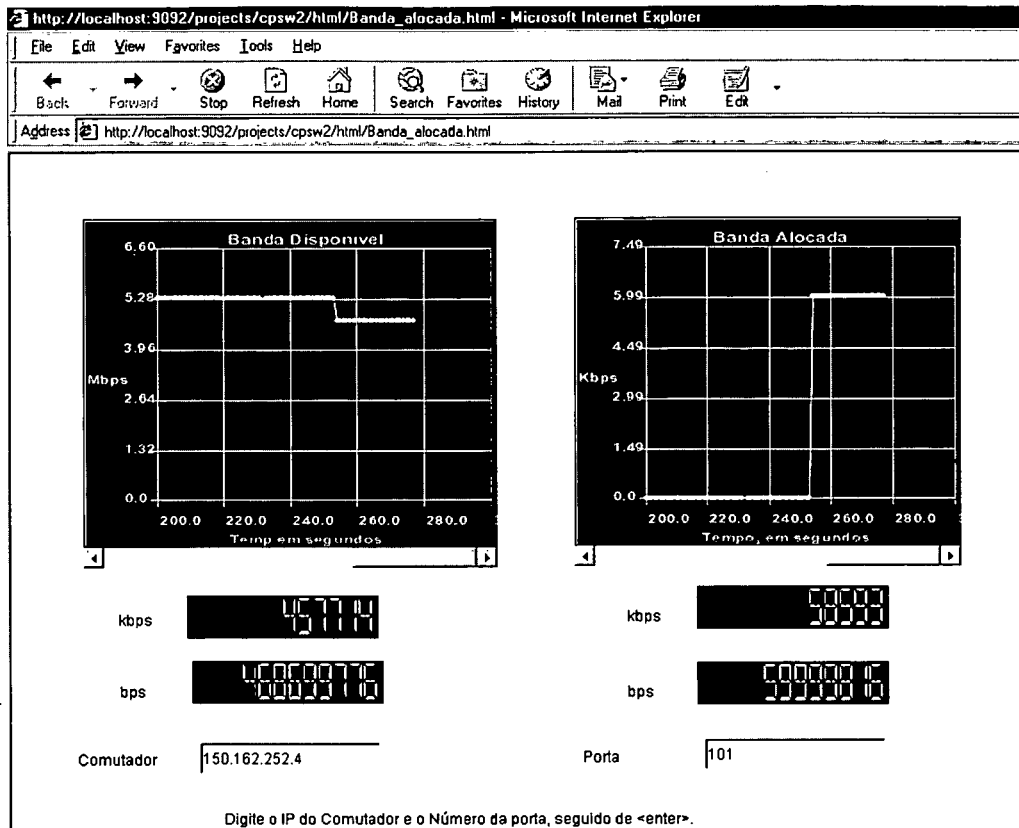
Status do switch	Sw_stat.class	Células transmitidas	Svc_cells.class
Status Módulos	Mod_stat.class	Banda Alocada/Utilizada	Svc_banda.class
Status comunicação Host/Link	Com_stats.class	Células Rejeitadas	Svc_rejeitos.class
Status das Fontes de Energia	Fontes.class	Endereços ATM dos sistemas finais	ATM_add.class
Temperatura do Switch	Temperatura.class	Clientes mais ativos	Dez_mais.class

**Tabela 8-2 - Módulos Funcionais do ATM**

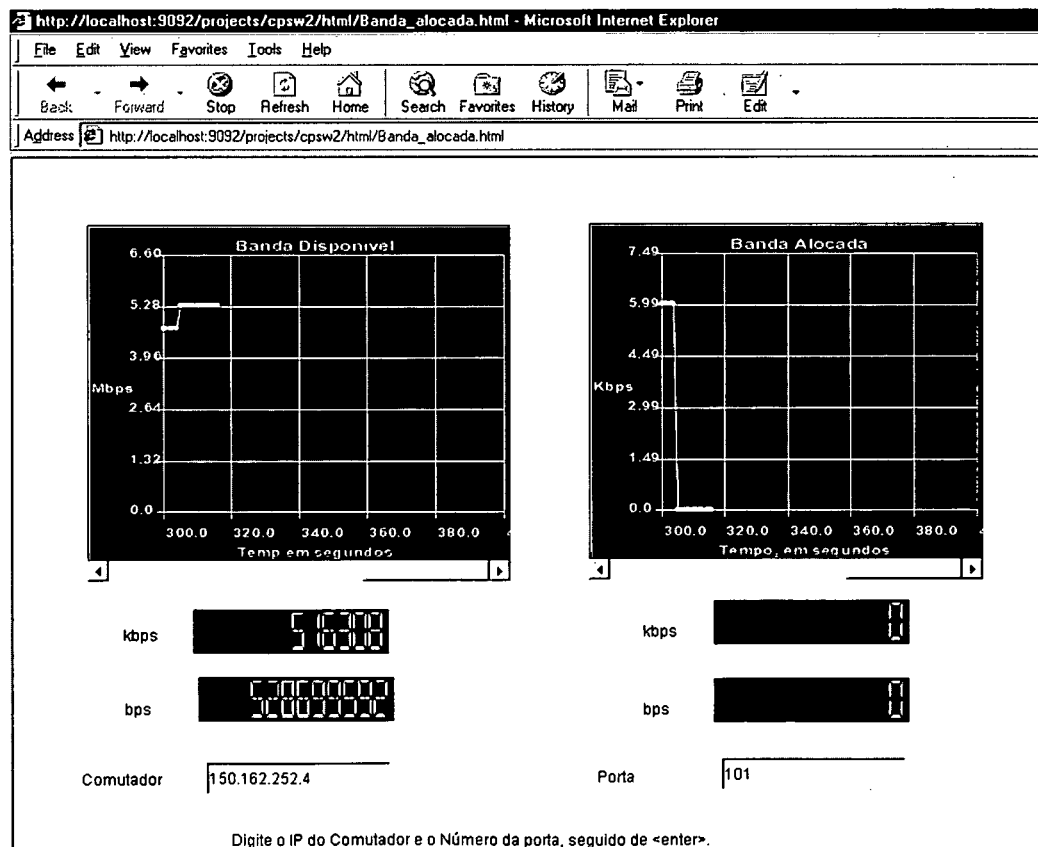
As figuras abaixo (Figura 8-6, Figura 8-7 e Figura 8-8) representam a alocação de banda em uma interface, quando solicitada uma conexão CBR por um usuário. No exemplo, o usuário solicitou uma conexão de 60000Kbps tanto na direção forward quanto na backward. Deve-se observar que a banda disponível está representada com uma ordem de grandeza a mais que a banda alocada.



**Figura 8-6 - Banda disponível e banda alocada em uma porta com tráfego UBR (bps)**



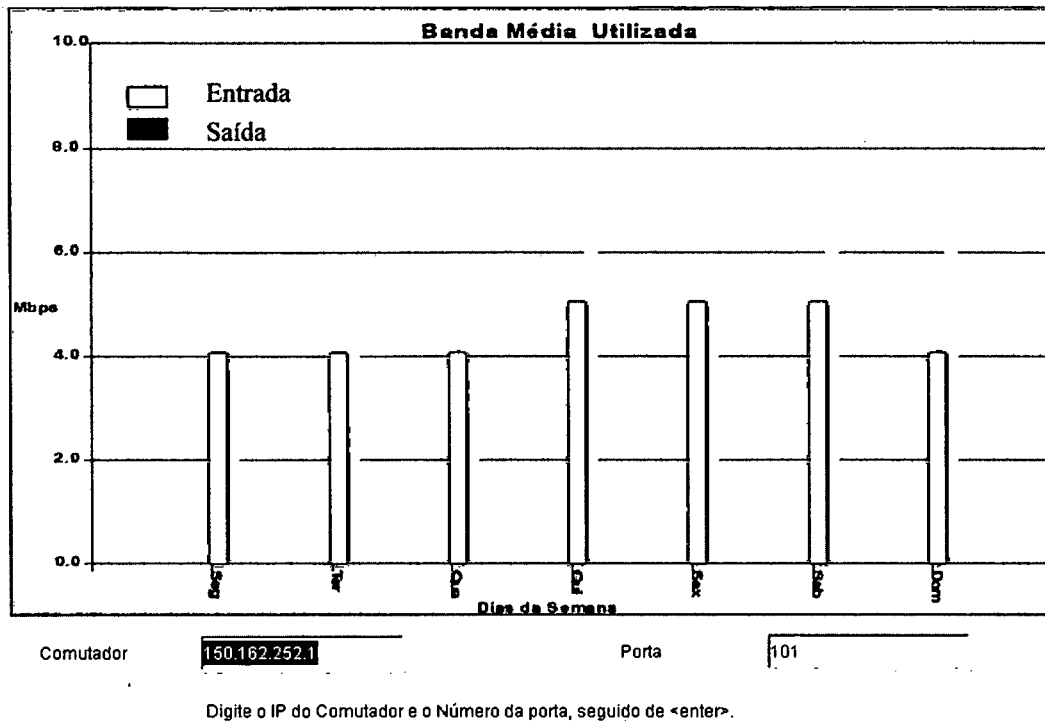
**Figura 8-7 - Banda disponível e banda alocada em uma porta com tráfego UBR mais uma conexão CBR de 50000 Kbps de taxa de pico (PCR).**



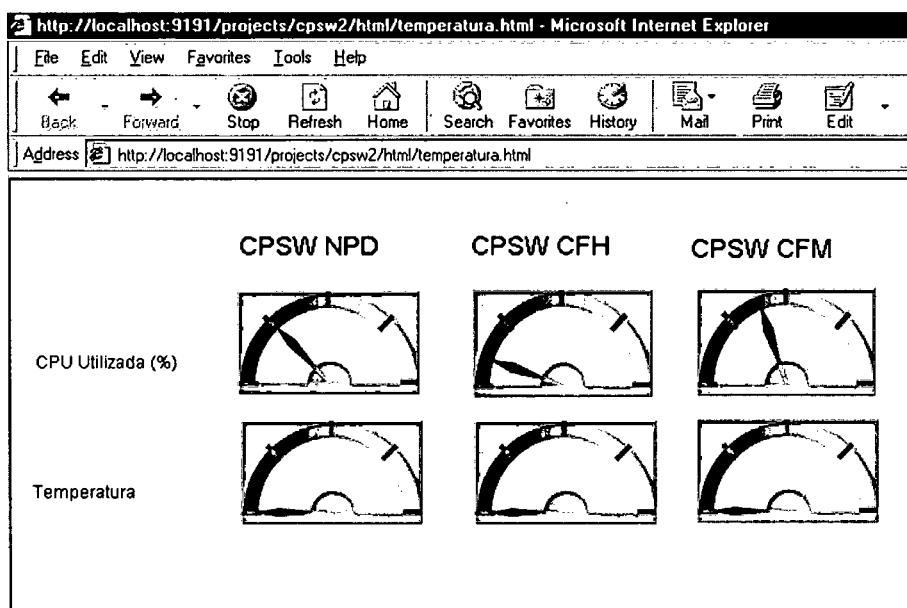
**Figura 8-8 - Banda disponível e banda alocada na mesma porta após o encerramento da conexão CBR**



A **Figura 8-9** mostra tela de visualização da utilização diária da porta 101 do comutador NPD. Pode-se perceber que a taxa de utilização das porta está bem abaixo da capacidade nominal de 622 Mbps.



**Figura 8-9 - Fluxo de entrada e saída de células**



**Figura 8-10 - Visualização da utilização das CPUs e temperatura ambiente nos comutadores**

### 8.5. Relacionamentos das unidades funcionais com as tabelas do DBMS

A seguir estão mostradas três tabelas do banco de dados. Na primeira, são armazenados os dados das portas físicas, como o número de células de entrada e saída e o número de conexões ativas. As demais tabelas são utilizadas para controle das conexões comutadas. A **Figura 8-12** mostra os campos onde são armazenadas as conexões UNI, e a **Figura 8-13** a estrutura dos campos para conexões genéricas, incluindo as de links PNNI e UNI.

Table: portas (dbo)

Key	Identity	Column Name	Datatype	Size	Nulls	Default
		sw_id	char	30	✓	
		porta	int	4	✓	
		erros_in	numeric	18,0	✓	
		unknow_in	numeric	18,0	✓	
		erros_out	numeric	18,0	✓	
		maxvccs	int	4	✓	
		banda_alocada	int	4	✓	
		banda_disponivel	int	4	✓	
		conex_ativas	int	4	✓	
		oper_status	char	4	✓	
		data	datetime	8	✓	
		desc_in	numeric	18,0	✓	
		desc_out	numeric	18,0	✓	
		erro_fisico	numeric	18,0	✓	
		cell_in	numeric	18,0	✓	
		cell_out	numeric	18,0	✓	
		max_conex	int	4	✓	
		admin_status	char	4	✓	

**Figura 8-11 - Estrutura de armazenamento para parâmetros de tráfego nas portas físicas dos comutadores**

Table: svc\_creation (dbo)

Key	Identity	Column Name	Datatype	Size	Nulls	Default
		sw_id	char	30		
		conex_id	char	10	✓	
		porta	int	4	✓	
		vpi	int	4	✓	
		vci	int	4	✓	
		atm_call	char	60	✓	
		atm_rec	char	60	✓	
		banda_alocada	numeric	18,0	✓	
		data	datetime	8	✓	

Ready | MN1\atmwtool | sa\dbo

Figura 8-12 - Estrutura da tabela utilizada para monitorar o estabelecimento de conexões comutadas UNI.

Table: svc\_trafego (dbo)

Key	Identity	Column Name	Datatype	Size	Nulls	Default
		svc_id	int	4	✓	
		celulas_in	int	4	✓	
		celulas_out	int	4	✓	
		descartes_in	int	4	✓	
		descartes_out	int	4	✓	
		sw_id	char	30	✓	
		porta	int		✓	

Ready | MYTILUS\atmwtool\svc\_trafego | sa\dbo

Figura 8-13 - Estrutura da tabela utilizada para monitorar o tráfego nas conexões comutadas, independente da sinalização (UNI, PNNI)

---

## 9. Resultados

---

Esta seção tem a finalidade de avaliar os resultados obtidos no presente estudo. Com às dificuldades inerentes ao tema, principalmente devido as ramificações e interdependências da tecnologia abordada, algumas metas iniciais ainda não puderam ser atingidas, e estão analisadas na seção 9.6 . A avaliação dos resultados está dividida em cinco tópicos, listados a seguir:

- Introdução
- Resultados Obtidos
- Análise Histórica Qualitativa
- Dificuldades Encontradas
- Propostas Futuras

### 9.1. *Introdução*

A necessidade de migração do backbone da redeUFSC para níveis compatíveis com as novas demandas de tráfego levou à opção pelas redes ATM. Devido a alta complexidade na implantação dessas redes, e ao caráter inovador da tecnologia, surgiu a motivação para efetuar-se o estudo da gerência de recursos. Até o momento, as maiores dificuldades para gerência dos comutadores centrais estão na ausência de algumas MIBs já padronizadas pelo ATM Forum e IETF.

Após a utilização plena da rede ATM, quando vários tipos de tráfego estiverem negociando conexões com a rede ATM, a ferramenta ATM será bastante útil na investigação das possíveis sobrecargas em função das topologias adotadas no backbone. Pretende-se, através do levantamento estatístico dos dados gerados pelo ATM, estabelecer padrões para rotas alternativas. Tais rotas deverão ser usadas caso a rede apresente pontos problemáticos em relação ao tráfego gerado pelas diferentes aplicações.

Espera-se que o impacto das novas aplicações multimídia seja minimizado através de um controle eficiente das variáveis pertinentes aos serviços CBR/VBR. Além

disso, pretende-se manter um alto grau de eficiência na utilização do meio, definindo políticas de prioridade de utilização do meio.

Finalmente, o ATRM-WTool está sendo usado para gerenciar os links ATM, tanto físicos como virtuais, bem como a disponibilidade de recursos no backbone da redeUFSC:

## **9.2. Resultados Obtidos**

Durante este estudo, conseguiu-se atingir vários dos objetivos propostos. A seguir estão descritas as situações de cada objetivo em termos de resultados alcançados.

### **9.2.1.1. Implantação do Backbone ATM da redeUFSC**

A rede ATM foi implantada, e está em funcionamento no nível do backbone e em algumas estações de trabalho (aproximadamente 40 Sistema final ATM não-proxy). Durante a fase de implantação foram realizados vários testes de interoperabilidade, análise dos diferentes links permanentes possíveis (ISS, IISP, PNNI). Este resultado conclui o objetivo 2.2.1, implantação da rede (**Figura 3-1**)

### **9.2.2. Monitoramento das conexões**

O objetivo 2.2.2, monitorar o estabelecimento de conexões deve centrar o foco na função de gerência de tráfego CAC [1] [53]. Uma das grandes vantagens das redes ATM está na possibilidade de utilizar-se a multiplexação estatística no canal de transmissão, através da qual pode-se superpor várias fontes (seção 7 da dissertação). Quanto maior o número de fontes superpostas, maior será o ganho da multiplexação estatística. Entretanto, o aumento indiscriminado de fontes de geração de sinal, que utilizam um canal compartilhado, pode levar a estados de congestionamento e/ou perda da QoS da rede, pela introdução de retardos e perdas de células. São necessários mecanismos de controle de tráfego que possam atender o maior número possível de conexões dentro dos parâmetros negociados no contrato de tráfego.

O Controle de Admissão de Conexões é uma das funções administradas no contrato de tráfego. O contrato de tráfego é uma negociação estabelecida pelas aplicações ATM para obter seus requerimentos de largura de banda e performance junto a rede. Essa negociação pode ser feita de duas formas [62] [65]:

- através de sinalização, quando se trata de uma conexão virtual comutada (SVC);
- pelo sistema de gerência da rede, tratando-se de uma conexão virtual permanente (VPC)

O presente estudo tem ênfase nas conexões comutadas (SVCs) por serem o tipo utilizado atualmente no ambiente da redeUFSC.

Existem aplicações de gerência desenvolvidas para atender especificamente o controle das conexões permanentes, utilizadas como única alternativa no início da tecnologia ATM, e agora, normalmente, para estabelecer circuitos entre redes privadas e operadoras comerciais. Uma dessas aplicações, denominada “UTOPIA”, foi desenvolvida pelo grupo Telematics Systems and Services Management (TSS) da Universidade de Twente, Holanda. Essa ferramenta está disponível na Web, inclusive com os códigos fontes (desenvolvidos em linguagem Java), na URL <http://www.snmp.cs.utwente.nl/nm/research/projects/utopia/release4.1>

Outros estudos tratam do estabelecimento das conexões permanentes e das comutadas [45] [51] [53] [61] [52] [54]. Esses estudos estão baseados na estrutura de gerência aprovada pelo IETF ainda em 1994, conhecida como AToM MIB, ou RFC 1695 [3]. Tal documento possui ênfase na gerência das interfaces ATM, possibilitando a configuração de conexões virtuais, principalmente as permanentes, e é considerado um superconjunto da MIB ILMI do ATM Forum.. Em fevereiro de 1999 foi proposto outro RFC (2515)[60] para substituir a MIB anterior. Este novo padrão já é suportado por alguns fabricantes, e permite um controle maior sobre as conexões comutadas. Neste estudo, para controlar as conexões virtuais, utilizou-se a AToM MIB 2 (RFC 2515) e a MIB proprietária da IBM para o comutador central CPSW modelo 8265 [30]. Através dessas estruturas de gerência, pode-se monitorar:

- o estabelecimento das conexões comutadas,
- consumo de recursos de cada uma delas,
- as estações geradoras da conexão,
- as estações destinatárias,
- tempo de duração de cada conexão

- algumas variáveis de controle de erros

O ambiente de gerência emite alarmes visuais para o gerente da rede quando ocorrerem os seguintes eventos:

- quando os recursos estão sobrecarregados, possibilitando ao gerente da rede cancelar conexões e canalizar os fluxos através de rotas alternativas, menos congestionadas.
- quando as taxas de erros das conexões desviarem-se de um padrão comportamental, que será estabelecido através de amostras armazenadas em um banco de dados.

Outras variáveis são monitoradas e estão descritas na seção 7.

### 9.2.3. Identificar os Níveis de QoS negociados em cada Conexão.

No momento em que efetua-se o presente estudo, apenas uma categoria de serviço está sendo requisitada pelas conexões da rede ATM no ambiente de estudos: é a classe UBR (Unespecified Bit Rate). Podemos estabelecer experimentalmente conexões do tipo ABR e CBR, através de comandos *atm\_ping* nos comutadores, como demonstrado na **Figura 8-6**, **Figura 8-7** e **Figura 8-8**. Entretanto, estabelecer uma conexão com reserva de banda não significa que o usuário utilize a banda alocada. Podemos comprovar isso pela estabilidade no fluxo de células mesmo após uma reserva de banda de 60 Mbps em um dos links dos comutadores.

As categorias utilizadas no nível de interface podem ser facilmente identificáveis pela variável *mib-2.atmMIB.atmMIBObjects.atmTrafficDescrParamTable.atmTrafficDescrParamEntry.atmServiceCategory (.1.3.6.1.2.1.37.1.5.1.10)*, da AToM MIB. As classes de QoS são retornadas pela variável *mib-2.atmMIB.atmMIBObjects.atmTrafficDescrParamTable.atmTrafficDescrParamEntry.atmTrafficQoSClass (.1.3.6.1.2.1.37.1.5.1.8)*

Para as conexões, conseguiu-se determinar as categorias de serviço e classes de QoS apenas para as conexões não efetivadas, que geram uma entrada em uma tabela de *log* na MIB proprietária da IBM. Todas as conexões sem sucesso que conseguiu-se detectar no ambiente de estudos aparecem nas interfaces dos roteadores MSS.

#### **9.2.4. Estudar as variáveis presentes nas MIBs definidas pelo ATM Forum e pelo IETF para a gerência de tráfego ATM.**

Duas MIBs padronizadas tratam do estabelecimento de conexões ATM: A ATOM MIB (RFC 1695 e RFC 2515, IETF) e MIB ILMI (ATMF). Esta última não está disponível para acesso pelas estações de gerência (NMS), conforme descrito na seção 7.4.

Pode-se ainda utilizar a MIB-II (RFC 1213) para monitorar as interfaces ATM, principalmente o grupo *interfaces*, que pode fornecer os dados de entrada e saída da *ifTable* como células. Para isso, basta dividir-se o número de octetos retornados pelos objetos por 53 bytes. [3] [45]. Foi necessário incluir nos estudos a MIB proprietária da IBM para conseguir-se dados sobre os endereços ATM de cada porta, e dos endereços fontes e destinos das chamadas. Também com essa MIB é possível monitorar o número de células transmitidas, recebidas e as células descartadas de cada conexão. Verifica-se que até o presente momento, não foi detectado nenhum descarte de célula nas conexões, provavelmente devido à pequena exigência da categoria de serviço que tem sido utilizada (UBR). O resultado deste estudo já foi descrito em detalhes na seção 7.

#### **9.2.5. Identificar os níveis críticos que os parâmetros referentes a cada classe de serviço podem atingir.**

Os níveis críticos que os parâmetros podem atingir são determinados através da análise histórica da rede. Somente com os dados históricos pode-se determinar um comportamento anormal, ou fora do padrão. Além dessa análise, a ferramenta está apta a emitir alarmes para o gerente humano, sempre que os recursos consumidos atingirem um nível que desvie-se da normalidade, ou aproxime-se dos níveis máximos permitidos pela capacidade atual da rede. Através do pacote estatístico Statistica V. 5.1 (<http://www.statsoftinc.com>), iniciou-se um levantamento estatístico básico da conduta de algumas variáveis, para a formação da *baseline*, como descrito na seção 1. Este levantamento inicial está demonstrado na seção 9.3.



### **9.2.6. Implementar a ferramenta de gerência ATRM-WTool**

A ferramenta foi implementada conforme descrito na seção 8, e vários parâmetros de tráfego estão sendo armazenados em uma base de dados relacional, utilizando-se como ferramentas principais:

- Adventnet Management Builder, V. 3.1, um ambiente de desenvolvimento totalmente escrito em java, que permite a utilização de várias classes de acesso ao protocolo SNMP.
- JDK 1.2, que permite o pleno funcionamento das classes SNMP do Adventnet Builder.
- MS-SQL Server V. 6.5, banco de dados relacional onde os dados estão sendo armazenados.

### **9.3. *Análise Histórica Qualitativa***

Efetuiu-se uma análise prévia sobre um período amostral de 15 dias. Algumas constatações preliminares puderam ser observadas, e estão demonstradas nas seções seguintes.

#### **9.3.1. Análise do fluxo de entrada/saída dos VCIs**

O fluxo de entrada e saída nos VCs dos comutadores apresentam taxas mais elevadas nos VCIs mais altos ( Figura 9-1 e Figura 9-2). Isso deve-se provavelmente ao fato de que os VCIs iniciais de cada VP estejam reservados para os canais de sinalização (seção 7.4).

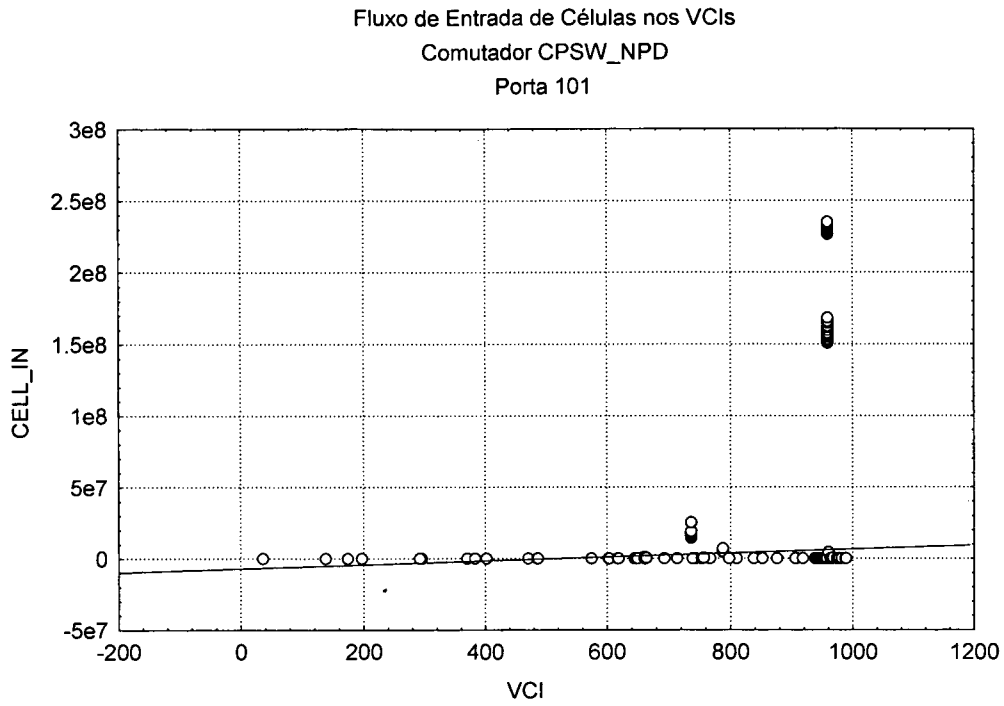


Figura 9-1 - Fluxo de saída nos VCs

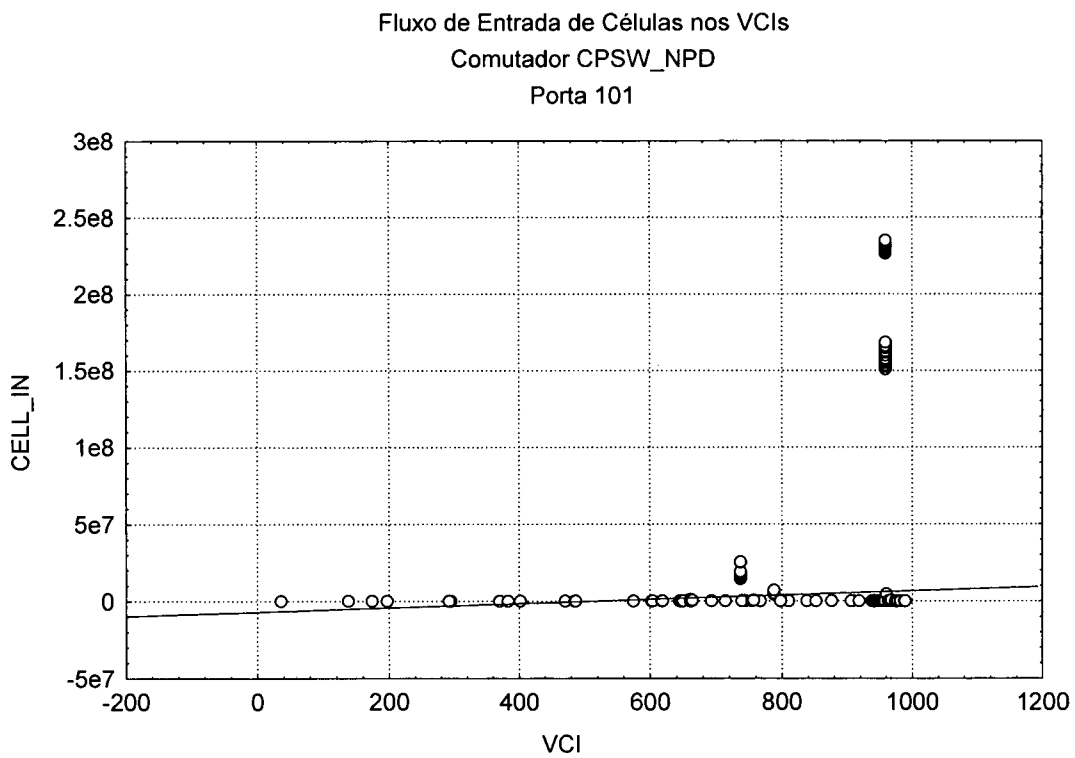
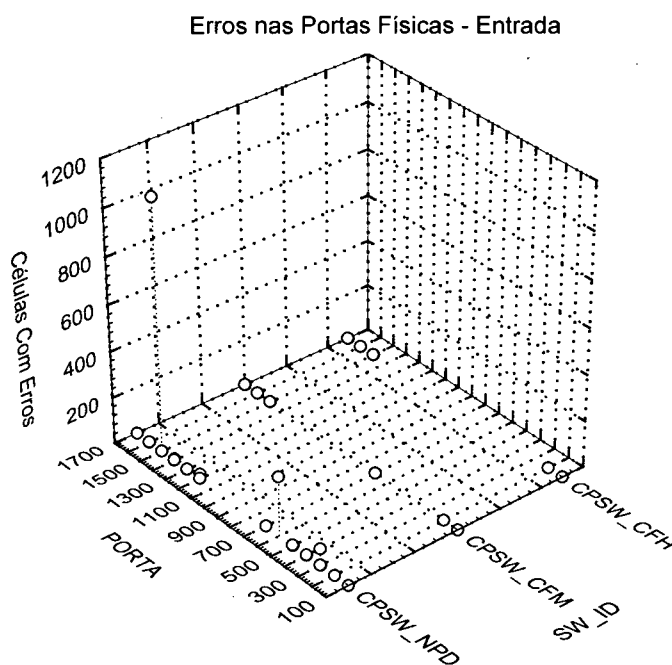


Figura 9-2 - Fluxo de Entrada nos VCs

### 9.3.2. Células com erros

Analizou-se também o comportamento de entrada de células com erros. Tais erros são descritos na seção 7.1, e representam o número de células descartadas devido a um erro incorrigível no HEC (*Header Check Error*). Pela análise descritiva, pode-se observar que as portas 1701 e 601 do comutador NPD apresentaram taxas significativas de erros durante o período amostrado (**Figura 9-3**).



**Figura 9-3 - Células com erros**

### 9.3.3. Número de Conexões ativas

O levantamento descritivo do comportamento das conexões ativas está demonstrado na Figura 9-4. Pode-se notar que as portas com maior número de conexões são aquelas correspondentes aos servidores de LES-BUS (Portas 701 para os comutadores NPD e CFM. O comutador CFH não possui servidor de LANE). O número de conexões se mantém em torno de 800 para a porta 701 do comutador NPD e 220 para o comutador CFM. Como o número máximo de conexões virtuais para essas portas é de 1024, mesmo no caso do comutador NPD existe uma margem de segurança.

NPD

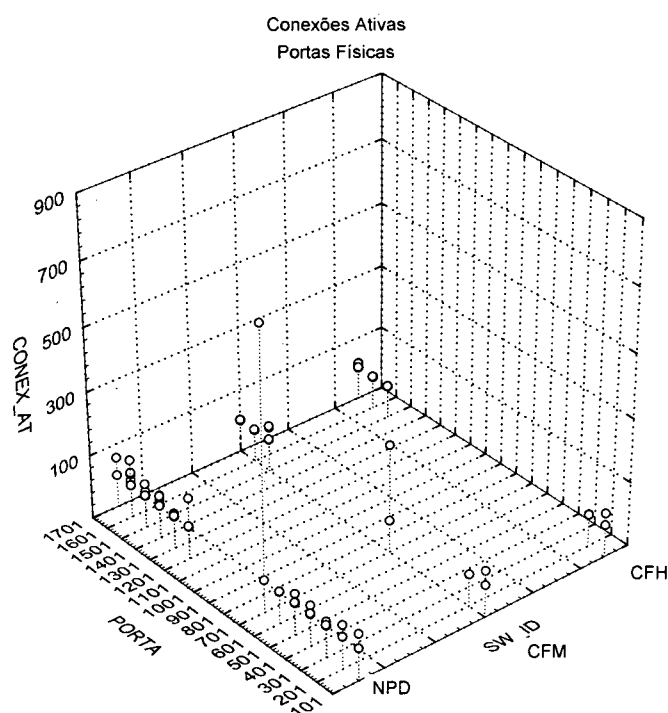


Figura 9-4 - Número de Conexões ativas

A análise descritiva do número de conexões ativas auxiliou na fase depuração da ferramenta ATM, demonstrando coletas erradas (número de conexões acima do permitido para as portas do *slot* 17 e períodos de latência (número nulo de conexões) na porta 701, responsável pelos serviços de LES, BUS e LECS (Figura 9-5). Sabemos experimentalmente que a cada novo cliente registrado em uma ELAN, as portas dos comutadores que possuem tais serviços devem abrir pelo menos mais 5 conexões ativas, responsáveis pela sinalização.

[Linha decorativa]

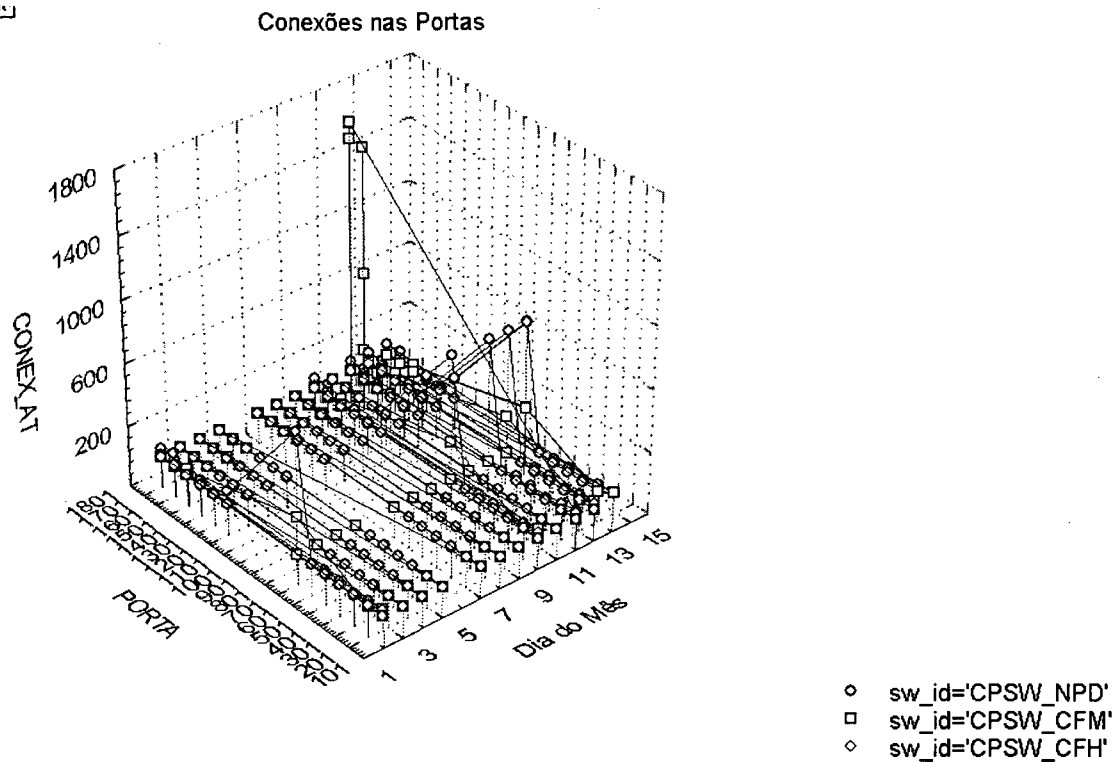


Figura 9-5 - Erro de amostragem durante a fase de depuração da ferramenta.

### 9.3.4. Frequência de saída de células

Na Figura 9-6 podemos notar que a maior frequência de fluxo de saída nas portas do comutador CPSW\_NPD corresponde a períodos de inatividade. Algumas dessas portas realmente apresentam *status* administrativo “down”, ou seja, não estão configuradas para operar.

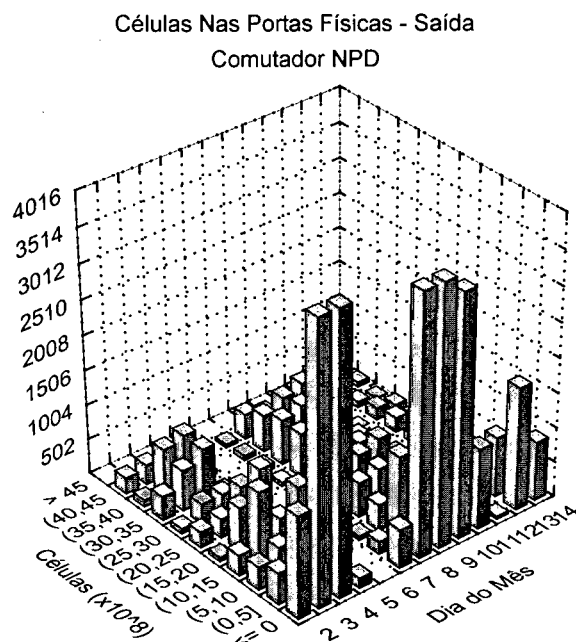


Figura 9-6 - Frequência de saída de células

#### 9.4. *Dificuldades Encontradas*

O SNMP é hoje o protocolo dominante para a gerência de redes ATM [65]. A grande maioria dos equipamentos e interfaces de rede (NICs) são fornecidas com agentes SNMP. Um grande número de MIBs têm sido padronizadas para gerenciar esses produtos. Em conjunto com as MIBs específicas dos fabricantes, já é possível uma gerência bastante flexível para as redes ATM. Entretanto, algumas limitações são salientes. A AToM MIB não possui todos os requerimentos para a gerência de uma rede ATM. Ela foi desenvolvida quando o ATM estava iniciando, quando existiam somente conexões permanentes. Mesmo sua nova versão não possui capacidades plenas sobre a rede. Embora já se possa ter alguma gerência sobre as conexões comutadas (SVCs), poucos fabricantes implementam a nova versão da MIB, especificada no RFC 2515 [60]. Os padrões de gerência ATM deveriam incorporar, atualmente, vários padrões de interoperabilidade já especificados para as demais operações da rede. No escopo deste estudo, falta principalmente um padrão de gerência que acompanhe as especificações de gerência de tráfego 4.0 [65]. Particularmente, os padrões de gerência deveriam suportar os novos tipos de conexões, SVC, e soft PVC.

Uma das maiores dificuldades encontradas foi a impossibilidade de medir-se os parâmetros pertinentes a performance e aos contratos de tráfego, como o CDV, devido a inexistência de aplicações e/ou agentes que pudessem gerar e monitorar as células RM (Resource Management). Deveria ser possível medir os intervalos de tempo no tráfego da rede, utilizando-se essas células de controle. Alguns fabricantes, como a CISCO, implementam agentes em seus comutadores centrais (atualmente o modelo Link Switch 1010) com recursos de monitoramento de tráfego bastante elaborados, como:

- Cisco-OAM-MIB, que mede parâmetros de retardo, tempos de *round-trip* (período necessário para que uma célula emitida pela interface transmissora chegue até a interface receptora e retorne ao ponto de partida);
- Cisco-ATM-TRAFFIC-MIB, que estende os parâmetros do modelo de tráfego da AToM MIB;
- Cisco-ATM-CONN-MIB, que trata das configurações das operações e monitoração das performances das conexões virtuais no que diz respeito a gerência de tráfego.
- Cisco-ATM-RM-MIB, que trata da gerência dos recursos do comutador.

A complexidade das redes ATM é outro fator que impõe dificuldades no estudo dessas redes, devido ao seu amplo espectro de abrangência no que diz respeito aos diferentes tipos de tráfego transportados.

A ausência de aplicações que utilizassem diretamente a camada ATM, ou passassem pela camada de adaptação com requerimentos diferentes da AAL5, utilizada pelo LANE, também imprimiu alguma dificuldade nos estudos. Utilizando a rede para transportar somente um tipo de tráfego, UBR, gerado pela Lan Emulation ou CIP, obtiveram-se parâmetros de comparação limitados. Some-se a isso o fato de ser o tráfego UBR aquele que menos exige garantias da rede, o que determina um nível praticamente nulo nos contadores de erros, descartes, congestionamento, alarmes, dos agentes SNMP dos comutadores.

## 9.5. *Conclusões*

No início dos estudos, pensou-se em monitorar todas as variáveis de tráfego constantes na especificação de gerência de tráfego do ATM Forum [65], principalmente aqueles referentes aos retardos (CTD, CDV). No entanto, tais variáveis são de difícil

monitoração. Conforme citado na seção 9.4, a maior parte dos fabricantes não implementa mecanismos de controle, seja através de agentes SNMP ou outro meio de acesso. Para essa finalidade, se precisariam adquirir ferramentas específicas para controle de tráfego ATM, do tipo "Sniffer", bastante onerosas. Tais ferramentas implementam coletas de dados nos canais de sinalização, possibilitando um controle de tempos de transmissão dentro das conexões. Algumas delas, apesar do preço elevado, conseguem levantar poucas informações além daquelas obtidas pela ATM (na Figura 9-7, está representada uma tela do ATM Sniffer® Network Analyzer, fabricado pela Network General®, [http://www.ngc.com/product info/sna/sna atm/atm.html](http://www.ngc.com/product%20info/sna/sna%20atm/atm.html)). Essa ferramenta possui funcionalidades adicionais que são de grande valor na gerência de tráfego, como é o caso do Optional Traffic Generator, que deve ser adquirido em separado. Outras, como ATM Sniffer Analyzer (<http://www.texascom.co.id/products/network/ngc/sniffer/atm.html>) fazem análise das sete camadas do modelo OSI e das três camadas inferiores do ATM, examinando inclusive as células.

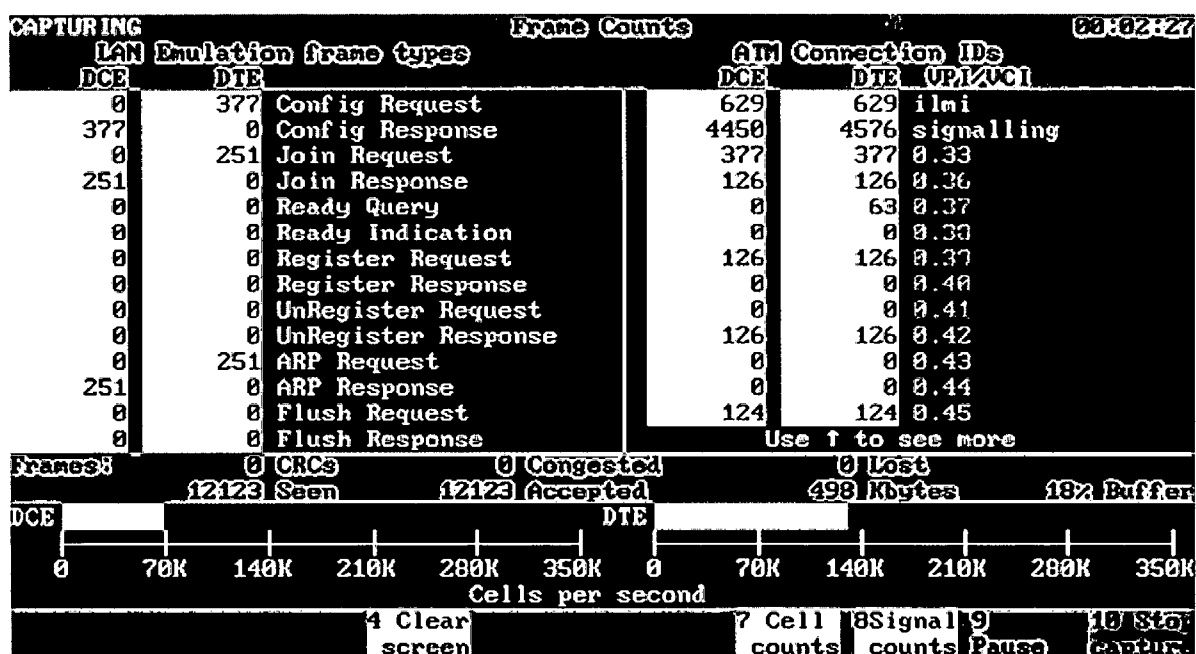


Figura 9-7 - Ferramenta Tipo "Sniffer" com funcionalidades para monitoramento de PVCs e SVCs

Apesar deste intuito inicial de medir-se os tempos de transmissão, congestionamento, controle de fluxo, os estudos realizados permitiram alcançar outros objetivos importantes que estavam previstos, como foram a implantação do backbone



---

ATM, a monitoração do estabelecimento de conexões e o controle de tráfego nas portas dos comutadores (descritos na seção 9.2).

### ***9.6.Sugestões para trabalhos futuros***

A realização deste estudo pode ser considerada um passo importante na direção do desenvolvimento de uma ferramenta para gerência de redes ATM. No momento em que se está concluindo este trabalho, pode-se constatar que existe um grande esforço dos organismos de padronização para eliminar as carências e diminuir a dificuldades da gerência de tráfego nas redes ATM. Como exemplo pode-se citar os documentos que estão sendo elaborados pelo IETF, através do AToM Working Group, demonstrados na lista de *draft standarts* da **Tabela 9-1**.

MIB	Descrição
ATM TC	Definições das convenções textuais e Object-Identities para gerência ATM
Supplemental	Definições de Objetos Gerenciados Suplementares para gerência ATM
Test	Definição de testes para gerência ATM
Next iteration AToM	Definição de objetos gerenciados para gerência ATM
History TC	Convenções textuais para módulos de MIB usando histórico de performance baseado em intervalos de 15 minutos
ATM History*	Objetos gerenciados para registro dos dados de performance baseado em intervalos de 15 minutos
Accounting Control	Objetos gerenciados para o controle de coleta e armazenamento de informações de contabilização para redes orientadas a conexão
Accounting Information	Informações de contabilização para redes ATM
* Não está claro quando este documento será padronizado	

**Tabela 9-1 – Lista atual de Drafts do AToM Working Group do IETF [45].**

Algumas MIBs já foram implementadas diretamente nos comutadores dos fabricantes de equipamentos ATM. Além disso, esses documentos possuem uma influência muito grande nos projetos das MIBs proprietárias. Segundo vários autores [9] [11] [15] [18] [20] [22] [11], deve-se esperar para breve a padronização definitiva desses documentos, e a conseqüente implementação nos agentes dos fabricantes.

Quando esse cenário estiver estabelecido, poderá gerenciar as redes ATM com um nível de refinamento muito grande, principalmente no que diz respeito às variáveis de tráfego. Enquanto os padrões não são implementados efetivamente, nossos trabalhos devem prosseguir no sentido de utilizar-se as APIs que estão sendo definidas pelo ATM Forum (Java API, <http://www.atmforum.com/atmforum/specs/specwatch.html>) e outras instituições de pesquisa, como por exemplo:

- [http://www.cse.ucsc.edu/~rom/projects/java\\_atm/presentation1.html](http://www.cse.ucsc.edu/~rom/projects/java_atm/presentation1.html)
- <http://www.sockets.com/winsock2.htm>
- <http://www.stardust.com/wsresource/winsoc2/atm.html>,

Em tal situação, se poderá implementar ferramentas que atuem diretamente no nível ATM. Como as APIs estão sendo descritas em java, a integração com a ATM está garantida.

A continuidade da ferramenta deve garantir ao gerente uma visão clara dos parâmetros do contrato de tráfego segundo as especificações do ITU-T e ATM Forum, demonstrados na Figura 9-8.

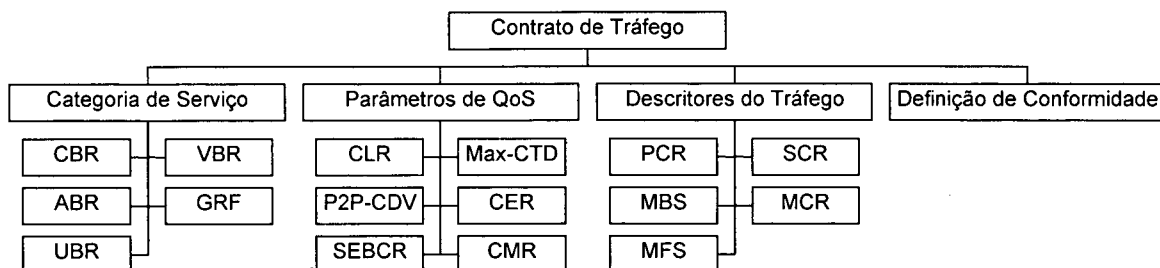


Figura 9-8 - Componentes do contrato de tráfego

A ferramenta deve ser trabalhada no sentido de incorporar também a gerência de contabilização, que, mesmo sem considerar-se os aspectos de tarifação para os usuários, seria interessante para o refinamento da política de controle de conexões.

No Anexo I podem ser encontrados estudos realizados sobre os objetos gerenciáveis da MIB ATM-SWITCHING-NODE-MIB v. 4.1.12. Tais objetos não estão sendo monitorados pela versão atual da ATM, mas foram disponibilizados para as versões futuras da ferramenta.

---

## 10. Bibliografia

---

- [1] **Abdalla, M. F. (1996):** Análise de Mecanismos de Controle Para Admissão de Conexão Para Redes ATM – Tese de Mestrado – Departamento de Engenharia Elétrica - COPPE/UFRJ.
- [2] **Abusamra, J.: (1998):** ATM Net Management: Missing Pieces Data Communications online tutorials. Na Internet: <http://www.data.com.tutorials/missing.html>.
- [3] **Ahmed, M.; Tesink, K. (1994):** Request for Comments: 1695 - Definitions of Managed Objects for ATM Management Version 8.0 using SMIv2
- [4] **Alexander, Peter; Carpenter, Kacey (1995):** ATM Net Management: A Status Report – In: In: Data Communications On The Web, Sept, 1995. Na Internet: [http://www.data.com/tutorials/ATM\\_Net\\_Management.html](http://www.data.com/tutorials/ATM_Net_Management.html)
- [5] **Baker, F.; Watt, J.(1993):** Request for Comments 1406 - Definitions of Managed Objects for the DS1 and E1 Interface Types
- [6] **Bernhardt, M. (1996):**Design and Implementation of a Web-Based Tool for ATM Connection Management – Master's Thesis – University of Stuttgart – Dept of Computer Science.
- [7] **Black, Uyles (1994):** Network Management Standards:SNMP, CMIP, TMN, MIBs, and Object Libraries – 2<sup>nd</sup>. Ed. - McGraw-Hill series on computer comunicatons.
- [8] **Brown, T.; Tesink, K. (1994):** Request for Comments 1595 - Definitions of Managed Objects for the SONET/SDH Interface Type
- [9] **Bruins, Barry (1996):** Some Experiences With Emerging Management Technologies – The Simple Times, Volume 4, Number 3, july, 1996 pg. 8-11. Na Internet : <http://www.simple-times.org/pub/simple-times/issues/4-3.html>
- [10] **Case, J.; Fedor, M.; Schoffstall, M.; Davin, J. (1990):** Request for Comments 1157 - A Simple Network Management Protocol (SNMP)
- [11] **Cekro, Z. (1998):** Management Information Base (MIB) Extensions for Asynchronous Transfer Mode (ATM) – Vrije Universiteit Brussel. Na Internet: <http://www.iihe.ac.be/scimitar/J1098/stc-98-05.html>

- 
- [12] Cooper, E. (1997): 1997: A Time of Consensus Building for ATM – Telecommunications, vol 31 n° 1 (p.18-22) - Jan 1997.
- [13] Coover, R. E. (1997): ATM Switches Artech-House, Norwood, USA.
- [14] Cox, T.; Tesink, K.(1993): Request for Comments: 1407 -Definitions of Managed Objects for the DS3/E3 Interface Type
- [15] Crosby, S. A. (1995): Performance Management in ATM Networks - St. John's College, University of Cambridge. Ph. D. Thesys.
- [16] Daines, B. (1997): The Future of Gigabit LANs. – Telecommunications, Vol 31, N° 1 – Jan 1997.
- [17] Deri, L. (1996): Surfin' Network Resources Across the Web – IBM Zurich Research Laboratory, Switzerland. Na Internet: <http://netman.cit.buffalo.edu/papers.html>
- [18] Dijk, A. P. van (1997): ATM Accounting Management M. Sc. Thesis – University of Twente – Tele-Informatics and Open System Group
- [19] Dobrowski, G. and Humphrey, M. (1996): ATM and Its Critics: Separating Fact from Fiction - Telecommunications, vol 30 N°11, (p.31-37) - nov 1996.
- [20] Dziong, Zbigniew (1997): ATM Network Resource Management – McGraw Hill.
- [21] Evans, E.; Rogers, D. (1995): Using java applets and CORBA for mult-user distributed applications. *IEEE Internet Computing*, May/June. 1995, <http://computer.org/internet/>.
- [22] Florissi, P. G. S. (1996): QoSME: QoS Management Environment - PhD Thesys, Graduate School of Arts and Sciences, Columbia University. Na Internet: <ftp://ftp.cs.columbia.edu/reports/reports-1995/cucs-036-95.ps.gz>
- [23] Gersht., A. and Lee, K.(1991): "A Congestion Control Framework for ATM Networks" – IEEE Journal in Selected Areas in Communications sep, 1991.
- [24] Giroux, N.; Sudhakar, G. (1998):Quality of Service in ATM Networks: State-of-the-Art Traffic Management – Prentice Hall, NJ.
- [25] Gunther, O.; Muller, R.; Schimdt, P.(1995): MMM: a web-based system for sharing statistical computing modules. *IEEE Internet Computing Mai/Jun 1995*
- [26] Händel, R.; Huber, M. N.; and Schröder, S. (1994): ATM Networks: Concepts, Protocols, Applications. 2<sup>nd</sup> ed. - Addison-Wesley.

- 
- [27] **Huang, A. , Moreland, C. and Wright, I. (1998):** *Advanced Traffic Management for Multiservice ATM Networks – Network Equipment Technologies, Inc. White Paper.*  
Na Internet:  
[http://www.net.com/repository/white\\_papers/adtm atm\\_wp/pdf/atmtraffic.pdf](http://www.net.com/repository/white_papers/adtm atm_wp/pdf/atmtraffic.pdf)
- [28] **Hyde, D. (1997):** *Web-Based Management: The New Paradigm for Network Management.* 3COM White Papers. Na internet:<http://www.3com.com/nsc.500627.html>
- [29] **IBM Corp. (1997):** *IBM PNNI Control Point (Switched Network Services)* White Paper - Zurich Research Lab
- [30] **IBM Corp. (1999):** *ATM-SWITCHING-NODE-MIB v. 4.1.2*
- [31] **ITU-T Recommendation I.356 (1996):-** *B-ISDN ATM layer cell transfer performance* - Telecommunication Standardization Sector of ITU - Series I: Integrated Services Digital Networks
- [32] **ITU-T Recommendation I.371 (1996):-** *Traffic Control and Congestion Control in B-ISDN* - Telecommunication Standardization Sector of ITU - Series I: Integrated Services Digital Networks
- [33] **ITU-T Recommendation Q.2931 (1995):** *B-ISDN – Digital Subscriber Signalling System No. 2 (DSS 2) – User-Network Interface (UNI) Layer 3 Specification For Basic Call/Connection Control* - Telecommunication Standardization Sector of ITU - Series Q: B-ISDN Application Protocols for Access Signalling.
- [34] **Jain, Raj. (1997) :** *ATM Networking: Issues and Challenges Ahead* – Dpt of Computer and Information Science. – The Ohio State University –Na Internet:  
[http://www.cis.ohio\\_state.edu/~jain/papers.html](http://www.cis.ohio_state.edu/~jain/papers.html)
- [35] **Krishnan, Kris & Fuller, Waine (1997):** *An Overview of MIBs for ATM Network Management* In: 53 Bytes – The ATM Forum Newsletter Vol 5:Num 4, Dezembro de 1997. Na Internet:  
[http://www.atmforum.com/atmforum/library/53bytes/current/article-53\\_12\\_97\\_03.html](http://www.atmforum.com/atmforum/library/53bytes/current/article-53_12_97_03.html)
- [36] **Kyas, Othmar. (1995):** *ATM Networks* – Ed. Internacional Thomson Computer Press, London, UK.
- [37] **McCloghrie, K.; Rose, M. (1990):** *Request For Comments 1156 - Management Information Base for Network Management of TCP/IP-based internets*

- 
- [38] McDysan, D. E. and Spohn, D. L. (1994): *ATM: Theory and Application* – McGraw-Hill series on computer communications.
- [39] McCloghrie, K. ; Rose, M. (1991): *Management Information Base for Network Management of TCP/IP-based internets: MIB II* - Request for Comments: 1213
- [40] Miller, Mark A. (1997): *Managing Internetworks With SNMP* - 2<sup>nd</sup>. Ed M&T Books
- [41] Mollenauer, J. (1996): *Rethinking ATM.* – Telecommunications, vol 30 n<sup>o</sup>8 (p.24) – Aug. 1996.
- [42] Mullaney, Patrick (1996): *Overview of a Web-Based Agent* - The Simple Times, Volume 4, Number 3, july, 1996 – pg. 11-16 .Na Internet : <http://www.simple-times.org/pub/simple-times/issues/4-3.html>
- [43] N.E.T. Withe Paper (1998): *Advanced Traffic Management for Multiservice ATM Networks.* Na Internet: [http://internet.net.com/techttop/adtatm\\_wp/home.html](http://internet.net.com/techttop/adtatm_wp/home.html)
- [44] Onvural, R. O. ; Cherukuri, R. (1997): *Signaling In ATM Networks* Artech-House, Norwood, USA.
- [45] Pan, H. (1998): *SNMP-Based ATM Network Management.* Artech-House, Norwood, USA.
- [46] Pras, A. (1995): *Network Management Architetures* – CTIT Ph. D-Thesis series N<sup>o</sup> 95-02 – Centre for Telematics and Information Technology nNetherlands.
- [47] Rayan, Gerald P. (1997): *ATM Traffic Management* - \_ATG's Communications & Networking Technology Guide Series. Na internet: <http://www.techguide.com>
- [48] Rose, M.; McCloghrie, K.(1990): *Request for Comments 1155 - Structure and Identification of Management Information for TCP/IP-based Internets*
- [49] Rose, Marshall (1994): *The Simple Book* – 2<sup>nd</sup> Ed. – Prentice Hall
- [50] Soares, L. F. G. et al (1995): *Redes de Computadores: Das LANs, MANs e WANs às Redes ATM.* Ed.Campus, São Paulo.
- [51] Sprenkels, R. A. M. (1996): *Management of ATM Networks* -M. Sc. Thesis – University of Twente – Tele-Informatics and Open System Group
- [52] Sprenkels, R. -Ed. (1997): *SURFNet4 ATM Management Project* University of Twente.  
<http://wwwsnmp.cs.utwente.nl/nm/research/projects/surfnet/final/1996/ctit-tr.pdf>
-

- 
- [53] Sprenkels, R.; Waaij, B. (1998): Offering an ATM SVC Service in a Production Environment – SURFNet ATM Managent Project, deliverable D3, v. #5. Un. Twente, The Netherlands.
- [54] Sprenkels, R.; Waaij, B.; Beijnum, B.; Pras, A. (1998): Management of Networks that Provides QoS Guarantees. CTIT, University of Twente, the Netherlands.  
<http://wwwsnmp.cs.utwente.nl/nm/>
- [55] Stallings, W. (1996): SNMP, SNMPv2, and RMON: Pratical Network Management 2<sup>nd</sup>. Ed. Addison Wesley
- [56] Stallings, W. (1995): ISDN and Broadband ISDN with Frame-Relay and ATM - Prentice Hall, New Jersey.
- [57] Stallings, W. (1998): High - Speed Networks: TCP/IP and ATM Design Principles - Prentice-Hall, Inc.- New Jersey.
- [58] Tanenbaum, A. S. (1997): Redes de Computadores Ed. Campus, Rio de Janeiro. Tradução da 3<sup>a</sup>. ed.
- [59] Tardy, G.; Treweel, K.; Sidhwa, F. G. (1998): IBM 8265 Nways ATM Campus Switch – IBM Red Books. Na Internet: <http://www.redbooks.ibm.com> )
- [60] Tesink, K. (1999): Request for Comments 2515 - Definitions of Managed Objects for ATM Management
- [61] Tesink, K.; Brunner, T. (1994): (Re)Configuration of ATM Virtual Connections with SNMP – The Simple Times, Vol 3, Num 2, Aug, 1994. Na Internet: <http://www.simple-times.org>
- [62] The ATM Forum (1994): ATM User-Network Interface Specification Version 3.1 – Prentice Hall, NJ.Na internet: <http://www.atmforum.com>
- [63] The ATM Forum Technical Committee (1996): ATM-FORUM TC-MIB definitions Na internet:<http://www.atmforum.com>
- [64] The ATM Forum Technical Committee (1996): Integrated Local Management Interface (ILMI) Specification Version 4.0. Na internet: <http://www.atmforum.com>
- [65] The ATM Forum Technical Committee (1996): Traffic Management Specification Version 4.0. Na internet: <http://www.atmforum.com>



- 
- [66] The ATM Forum Technical Committee (1997): *Remote Monitorin MIB Extensions for ATM Networks* – *af-nm-test-0080.000*  
Na Internet: <http://www.atmforum.com>
- [67] Todd, Stephen (1997): *HMMP Overview* - Internet Draft. Na Internet:  
<http://wbem.freerange.com/wbem/draft-hmmp-overview-03.txt>
- [68] Townsend, R. L. (1995): *SNMP Application Developer's Guide*. International Thomson Publishing, Inc. NY.
- [69] Worden, D. J. (1994): *Sybase Developers Guide* – SAMS publishing, Indianapolis.
- [70] Zalloua, Paul (1996): *ATM Inverse Multiplexing: Time for IMA*. In: Data Communications On The Web, Sept, 1996. Na Internet:  
<http://www.data.com/tutorials/ima.html>

## 11. Siglas

<b>Sigla</b>	<b>Significado</b>
AAL	ATM Adaptation Layer
ABR	Available Bit Rate
ABT	ATM Block Transfer (ITU-T)
ATC	ATM Transfer Capability
B-ISDN	Broadband ISDN
BUS	Broadcast and Unknow Server
CAC	Connection Admission Control
CBR	Constant Bit Rate
CDV	Cell Delay Variation
CDVT	Cell Delay Variation Tolerance
CER	Cell Error Ratio
CET	Cell Emission Time
CLP	Cell Loss Priority bit
CLR	Cell Loss Ratio
CMR	Cell Misinsertion Rate
CRC	Cyclic Redundancy Check
CRE	Cell Reference Event
CSMA/CD	Common Shared Media Access/Colision Detect
CTD	Cell Transfer Delay
DBR	Deterministic Bit Rate (ITU-T)
DQDB	Distributed Queue Dual Bus
EFCI	Explicit Forward Congestion Indication
EPD	Early Packet Discard
FCAPS	Fault, Configuration, Accounting, Performance, Security
GCRA	Generic Cell Rate Algorithm
HEC	Header Check Error
HMMP	HyperMedia Management Protocol
HTML	Hyper Text Markup Language
HTTP	Hyper Text Transmission Protocol
IETF	Internet Engineering Task Force
ILMI	Integrated Local Management Interface
IMP	International Measurement Point
ISDN	Integrated Services Digital Network
LANE	LAN Emulation
LE ARP	LAN Emulation Address Resolution Protocol
LEC	LAN Emulation Client
LECS	LAN Emulation Configuration Server
LES	LAN Emulation Server

LUNI	LAN Emulation User Network Interface
MAC	Media Access Control
MBS	Maximum Burst Size
MCR	Minimum Cell Rate
MCSN	Monitoring Cell Sequence Number
MFS	Maximum Frame Size
MP	Measurement Point
MPT	Measurement Point at $T_B$
NNI	Network-to-Network Interface
NP	Network Performance
NPC	Network Parameter Control
NPC	Network Parameter Control
OAM	Operations and Maintenance
OAM&P	Operations, Administration, Maintenance and Provisioning
OSI	Open System Interconnection
P2P-CDV	Peak-to-Peak Cell Delay Variation
PCR	Peak Cell Rate
PDH	Plesiochronous Digital Hierarchy
PDU	Protocol Data Unit
PL	Physical Layer
PM	Performance Monitoring
PNNI	Private Network-Network Interface
PT	Payload Type
PVC	Permanent Virtual Circuit
QoS/ QOS	Quality of Service
RTT	Round Trip Time
SAP	Service Advertising Protocol
SBR	Statistical Bit Rate
SCR	Sustained Cell Rate
SDH	Synchronous Digital Hierarchy
SECBR	Severely Errored Cell Block Ratio
SNMP	Simple Network Management Protocol
SSN	Switching / Signaling Node
STM	Synchronous Transfer Mode
SVC	Swiched Virtual Circuit
T	Nominal Cell Interarrival time
TCP/IP	Transport Control Protocol
$T_{max}$	Timer for Declaring a Cell Lost
TUC	Total User Cell
UBR	Unspecified Bit Rate
UNI	User Network Interface
UPC	Usage Parameter Control
UPC	Usage Parameter Control
VC	Virtual Channel
VCC	Virtual Channel Conection

---

VCI	Virtual Channel Identifier
VP	Virtual Path
VPC	Virtual Path Connection
VPI	Virtual Path Identifier

---

## Anexo I.

---

Variáveis da MIB proprietária IBM-ATM-SWITCHING-NODE-MIB v. 4.1.12 que foram selecionadas devido a sua importância para gerência de parâmetros de tráfego e recursos, mas não são monitoradas na ferramenta ATM, em sua versão atual.

---

---

◆ **Grupo *atmSvcLogTable***

Esta tabela contém uma lista das últimas SVCs que foram completadas neste nó ATM sob condições excepcionais. A tabela não contém entradas para as conexões completadas normalmente

Objeto *atmSvcLogIndex*: Um valor que identifica a entrada nessa tabela, determinada pelo agente SNMP local. Esse valor é estabelecido em cada chamada SVC nova, ou quando ocorre uma requisição da interface parceira da conexão. Este índice é alocado em ordem decrescente, de forma que uma requisição *get next* na tabela recupera primeiro a última chamada.

Objeto *atmSvcLogInterfaceIndex*: O valor *ifIndex* da interface ATM usada pela SVC que originou a entrada na tabela.

Objeto *atmSvcLogCallingNumber* : O endereço ATM da interface solicitante<sup>12</sup> da chamada, constante no IE (Elemento de Informação) transportado. Este elemento está presente na mensagem de chamada para o estabelecimento da conexão (*Call Setup Message*).

Objeto *atmSvcLogCalledNumber*: Contém o endereço descrito acima, apenas para a interface receptora da chamada.

Objeto *atmSvcLogCreationTime* : Data e hora do estabelecimento da conexão.

Objeto *atmSvcLogTime*: Data e hora na qual a conexão foi encerrada, ou seja, retirada da tabela.

Objeto *atmSvcLogClearCause*: O motivo que levou ao encerramento da conexão. Esse valor normalmente tem sido igual a 3. Segundo a MIB da IBM [30], esses valores estão descritos na especificação "ATM Forum/93-265R5 Signalling Specification Draft - Apr. 14, 93."

---

<sup>12</sup> Neste estudo, denominamos "interface solicitante" a extremidade da conexão que solicitou a chamada, e "interface solicitada" a extremidade que recebeu a solicitação da conexão. Esses termos foram traduzidos de forma arbitrária dos originais em inglês "*calling party*" e "*called party*", respectivamente.

---

Objeto *AtmSvcLogForwardQOS*: A QoS requisitada para transmissão nesta chamada. Os valores pode ser os constantes na Tabela I-1. Esta variável possui o status de “*deprecated*”, e provavelmente deve ser substituída pela “categoria de serviço” (Tabela I-2).

Classe	Valor de retorno
A	1
B	2
C	3
D	4

**Tabela I-1 - Valores retornados pelas variáveis *AtmSvcLogForwardQOS* e *AtmSvcLogBackwardQOS***

Objeto *atmSvcLogBackwardQOS*: A QoS requisitada para recepção nesta chamada (Tabela I-1).

Objeto *atmSvcLogForwardBW*: A largura de banda solicitada por esta conexão para a transmissão dos sinais. Os valores retornados não estão especificados em termos de unidades, mas para todas as conexões são igual a 89.

Objeto *atmSvcLogBackwardBW*: A largura de banda solicitada por esta conexão para a recepção dos sinais. Os valores retornados não estão especificados em termos de unidades, mas para todas as conexões são igual a 89.

Objeto *atmSvcLogServiceCategory*: A categoria de serviço requisitada pela conexão. Os valores retornados estão descritos na **Tabela I-2**. Todas as conexões com entradas nessa tabela têm retornado o valor 6, correspondente a categoria UBR.

Categoria de serviço	Valor de retorno
Outra	1
CBR	2
rtVBR	3
nrtVBR	4
ABR	5
UBR	6

**Tabela I-2 - Valores de retorno para a Categoria de serviço**

Objeto *atmSvcLogClearLocation*: A fonte que originou a desconexão desta chamada. Os valores de retorno estão descritos na Tabela I-3. Odoos os valores retornados até o momento correspondem a “rede privada servindo usuário local” (1).

Fonte	Valor de retorno
Usuário	0
Rede privada servindo o usuário local	1
Rede pública servindo o usuário local	2
Rede transitória	3
Rede pública servindo o usuário remoto	4
Rede private servindo o usuário remoto	5
Rede Internacional	7
Rede intermediária, entre duas redes	10

**Tabela I-3 - Valores de retorno para o local de solicitação de encerramento da conexão**

#### ◆ Grupo *AtmQ2931StatsEntry*

Este grupo possui as entradas da tabela *atmQ2931ConfTable*. Cada entrada corresponde a um referência de interface ATM e um canal de sinalização. O canal de sinalização possui um índice único em cada interface, definido pelos valores de VPI e VCI alocados. A interface ATM corresponde a interface virtual definida pelo *ifType* da MIB II como 53 (Proprietária Virtual), segundo referenciado na Tabela 7-1.

Objeto *atmQ2931StatsIndex*: O valor do *ifIndex* da interface ATM o qual, juntamente com os valores de VPI e VCI do canal de sinalização, servem como identificador único da entrada na tabela. Exemplo de retorno: **1.0.0: 1**, significando que a interface é igual a 1, o *vpi*=0 e *vci*=0. O retorno é =1, correspondente ao *ifIndex* da interface 1.

Objeto *atmQ2931StatsVpi*: O valor do *Vpi* que, em conjunto com o valor de *Vci* retornado por *atmQ2931StatsVci*, define o canal de sinalização utilizado por esta entrada na tabela. Usualmente, existe um canal de sinalização por interface, definido por *Vpi*=0, *Vci*=5.

Objeto *atmQ2931StatsVci*: O valor do *Vci* que, em conjunto com o valor de *Vpi* anterior, define o canal de sinalização.

Objeto *atmQ2931OutCallAttempts*: Retorna o número de chamadas de conexão a partir desta interface, incluindo as que foram aceitas e as rejeitadas.



---

Objeto atmQ2931OutCallInProgress: Retorna o número de chamadas que estão sendo efetuadas a partir desta interface, no momento da consulta.

Objeto atmQ2931OutCallFailures: Retorna o número de chamadas, a partir desta interface, que foram desconectadas por razões diferentes de uma ação iniciada ou pelo usuário ou pelo DTE (*Data Terminal Equipment*).

Objeto atmQ2931InCallAttempts: Retorna o número de chamadas recebidas pela interface, incluindo as que foram aceitas e as rejeitadas

Objeto atmQ2931InCallInProgress: Número de chamadas que estão sendo recebidas pela interface no momento da consulta.

Objeto atmQ2931InCallFailures: Número de chamadas rejeitadas pelo receptor.

#### ◆ **Grupo nbrTable**

Esta tabela contém características básicas dos dispositivos ATM adjacentes conectados ao comutador.

Objeto nbrEntry.nbrIpAddress1: Um ds endereços IP do agente SNMP ATM do nó conectado a esta porta/slot. Quando não disponível, o agente retorna 0.0.0.0.

Objeto nbrAtmAddress: O endereço ATM do dispositivo conectado a esta porta ou slot. Quando não disponível, o agente retorna uma string nula.

Objeto nbrName: O valor da variável *sysName* da MIB-II é retornado pelo dispositivo conectado a esta porta. Quando não disponível, uma string nula é retornada.

#### ◆ **Grupo vcXConnectTable**

Esta tabela contém as conexões cruzadas (*cross-connections*) configuradas no comutador para todos os links virtuais (VCLs), tanto para PVCs quanto para SVCs.

Objeto vcXInIndex: O número da interface desta porta ATM, onde a conexão está chegando. Um comando SNMP *Get* na porta 402 pode retornar um valor do tipo:

---

---

402.0.438.1702.0.175: 402
---------------------------

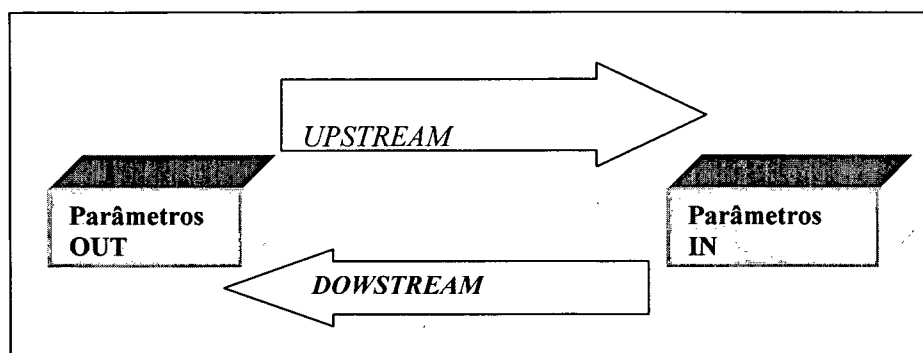
(1) (upstream)

A mesma conexão na porta 1702 retorna o seguinte valor:

1702.0.175.402.0.438:1702.
----------------------------

(2) (downstream)

Esse índice pode ser interpretado como “a conexão cruzada da porta 1702, com o vpi=0 e vci=175, para a porta 402, vpi=0 e vci= 438”. A direção da conexão é dada pela variável *vcXDirection* descrita logo a seguir. Quando a direção é *Downstream*, foi originada na interface que possui os parâmetros de entrada (*In*) para a interface com os parâmetros de saída (*Out*). No exemplo acima, seja, a conexão foi originada na interface 1702. Pode-se chegar a essa conclusão pelo fato de que o retorno (1) aponta a interface 402 com direção *Upstream*, ou seja, na extremidade contrária ao valor de retorno do comando SNMP Get. Genericamente, pode-se demonstrar essa situação através da **Figura I-1**.



**Figura I-1- Direção do fluxo da conexão**

A conexão utilizada como exemplo pode ser resumida através da Tabela I-4.

<b>IfIndex</b>	<b>vcXInIndex:</b>	<b>vcXOutIndex:</b>	<b>vcXDirection</b>
402	402.0.438.1702.0.175:402	402.0.438.1702.0.175:1702	Upstream
1702	1702.0.175.402.0.438:1702	1702.0.175.402.0.438:402	Downstream

**Tabela I-4 - Exemplo de Conexão Cruzada**

Objeto vcXInVpi: O valor do Vpi para esta conexão.

Objeto vcXInVci: O valor do Vci para esta conexão.

Objeto vcXOutIndex: O número da interface para esta porta ATM. Seguindo o mesmo exemplo citado na variável *vcXInIndex*, teria-se como exemplo de retorno:

**1702.0.175.402.0.438:402**

O que significa dizer que existe uma conexão cruzada entre a porta 1702 (Vpi=0, Vci=175) e a porta 402 (Vpi=0, Vci=438). Para saber-se qual interface chamou a conexão, usa-se a variável *vcXDirection* (Tabela I-4 e Figura I-1).

Objeto vcXOutVpi: O valor do Vpi para esta conexão.

Objeto vcXOutVci: O valor do Vci para esta conexão.

Objeto vcXType: Define se a conexão cruzada faz parte de uma conexão unicast ou multicast.

Objeto vcXDirection: Esta variável identifica se o fluxo da conexão cruzada é *Upstream* ou *Downstream*, do ponto de vista da origem. *Downstream* significa que a

conexão foi estabelecida a partir dos parâmetros *In* (interface, vpi, vci) até a interface, vpi e vci retornadas pelas variáveis *out* (Tabela I-4 e Figura I-1). Para uma conexão multicast, em particular, *Downstream* significa que a iniciação da chamadas (a origem), está na interface retornada pela variável *vcXInIndex*.

#### ◆ *Grupo interfaceTable*

Para cada porta ATM, esta tabela mapeia uma interface da MIB II, com seu slot físico e número da porta no slot. Cada entrada na tabela corresponde a uma porta que pertence a um módulo ATM do comutador.

##### Objeto interfaceAdminState:

O estado administrativo dessa porta. Quando configurada para *disabled*, nenhum tráfego ATM pode passar pela porta. Todas as conexões (PVCs e SVCs) são apagadas.

Quando configurada para *wrap-reply*, essa interface é colocada em estado de “repetidora” (*wrapped*) de forma que todo tráfego recebido pela linha conectada a porta é remetido de volta. Se o estado da interface não for alterado, ela volta ao estado *disabled* após 60 segundos.

Quando o estado é configurado para *wrap-far-end*, a interface na outra extremidade do link foi colocada no estado de “repetidora”. O estado de *wrap-reply* só pode ser configurado em interfaces 155 e Wan. O estado de *wrap-far-end* não pode ser configurado, somente pode ser verificado (*read-only*).

Quando o estado *reset-to-default* é configurado, a porta é desabilitada e seus atributos são reinicializados com os valores default. Este estado não pode ser lido (*read*). O estado *reset-to-default* aplica-se somente para agentes do modelo 8260 v.3.0 ou para os agentes 8265 versão 2.0 ou mais recentes

##### Objeto interfaceOperState: Representa o estado operacional desta porta:

- *unknown*: O estado da porta é desconhecido, o que pode ser causado por um erro no módulo do comutador.

- 
- *disabled-nosignal*: Nenhuma atividade está sendo detectada na camada física enquanto a porta está em estado de *disabled*
  - *disabled-idle*: A porta está desabilitada, mas foi detectada atividade na camada física, proveniente do dispositivo remoto.
  - *failing*: Ocorreu um erro interno grave quando a porta foi habilitada.
  - *no signal*: Nenhuma atividade foi detectada na camada física quando a porta foi habilitada.
  - *idle*: Foi detectada atividade no dispositivo remoto pela camada física. A porta local esta em estado *enabled*.
  - *in-service*: O dispositivo remoto respondeu aceitando as requisições do *pooling* ILMI e do registro da SVC.
  - *in-service-no-address-registration*: O dispositivo remoto respondeu aceitando as requisições do *pooling* ILMI, mas rejeitou o pedido de registro de prefixos ATM. Somente são suportadas conexões PVC.
  - *misConfigured*: Valor retornado quando uma porta PNNI está conectada a uma UNI privada ou pública.
  - *wrongNetworkPrefix*: Os comutadores em cada extremidade de um link PNNI possuem prefixos de rede incompatíveis (os primeiros 12 bytes possuem valores diferentes)
  - *wrongNodeNumber*: retornado quando os comutadores nas extremidades de um link PNNI possuem o mesmo número de nó ATM (byte 13).
  - *failing-line*: Esta porta foi habilitada (estado *enable*) e um sinal inválido foi detectado na linha.
  - *disabled-failing*: A porta está em estado *disable* e uma anomalia foi detectada, tanto interna quanto externa.
  - *wrap-no-signal*: A porta está com estado interno de “repetidora”(wrapped) de forma que todo o tráfego recebido na linha conectada é retornada sem nenhuma alteração para a linha. Nenhuma atividade foi detectada na camada física.
  - *wrap-idle*: A porta está em estado interno *wrapped*. Sinais válidos estão sendo detectados na linha.
-

- 
- *wrap-failing-internal*: Uma falha interna foi detectada quando a porta foi colocada em estado de *wrap-reply*. O estado atual da porta está indefinido.
  - *wrap-failing-line*: A porta está em estado de *wrapped* e um sinal inválido foi detectado na linha.
  - *idle-no-bandwidth*: A porta está em estado de *enabled* e foi detectada atividade no dispositivo remoto, mas não existe largura de banda para operar na porta com sua configuração atual. Este estado se aplica aos links PNNI.
  - *idle-internal-error*: A porta está habilitada, e foi detectada atividade no dispositivo remoto, mas ocorreu um erro interno durante a verificação da configuração da porta.
  - *disabled-no-bandwidth*: Não foi possível habilitar a porta, porque não existe largura de banda suficiente para operar na configuração atual. Este estado se aplica a links PNNI.
  - *wrap-far-end-no-signal*: A porta está no estado *wrap-far-end*, ou seja, o ponto final remoto da linha está *wrapped*, e nenhum sinal foi detectado na linha.
  - *wrap-far-end-idle*: A porta está no estado *wrap-far-end* e um sinal válido foi detectado na linha.
  - *wrap-far-end-failing*: A porta está no estado *wrap-far-end* e um erro interno foi detectado.
  - *wrap-far-end-failing-line*: A porta está no estado *wrap-far-end* e um sinal inválido foi detectado na linha.
  - *insufficient-connection-handles*: A porta está habilitada e foi detectada atividade, mas existem muitas conexões ativas para o módulo, e as conexões de controle para esta porta não puderam ser estabelecidas.
  - *invalid-remote-vpi-vci-range*: A porta está habilitada e foi detectada atividade na linha, mas o número de bits para o vpi ou vci retornado pelo dispositivo conectado não é suportado pelo sub-sistema ATM
  - *control-vpi-already-used*: A porta está habilitada, e foi detectada atividade na linha, mas o valor de vpi designado para os canais de ILMI, sinalização e roteamento já estão em uso por uma VC
-

- *missing-signalling-version*: A porta está habilitada e foi detectada atividade na linha, mas o ILMI está desabilitado e a versão de sinalização está configurada para “automática”.

Sempre que uma porta estiver em estado disabled, somente pode estar também em um dos estados a seguir: unknown, disabled-failing, disabled-nosignal, disabled-idle ou disabled-no-bandwidth.

Os valores retornados estão representados na **Tabela I-5**.

Estado operacional da porta	Retorno
unknown	1
disabled-nosignal	2
disabled-idle	3
no-signal	4
idle	5
in-service	6
in-service-no-address-registration	7
failing-internal	8
misConfigured	9
wrongNetworkPrefix	10
wrongNodeNumber	11
disabled-failing	12
failing-line	13
wrap-no-signal	14
wrap-idle	15
wrap-failing-internal	16
wrap-failing-line	17
idle-no-bandwidth	18
idle-internal-error	19
disabled-no-bandwidth	20
wrap-far-end-no-signal	21
wrap-far-end-idle	22
wrap-far-end-failing	23
wrap-far-end-failing-line	24
Insufficient-connection-handles	25
Invalid-remote-vpi-vci-range	26
Control-vpi-already-used	27
Missing-signalling-version	28

**Tabela I-5- Valores retornados pela variável *interfaceOperState***

Objeto interfaceAtmAccess: O tipo de acesso ATM oferecido nesta porta:

- privateUni: UNI privada
- privateNetwork: PNNI
- publicUni: UNI pública. O protocolo ILMI pode rodar em portas com UNI pública. Um túnel VP pode ser criado em tais portas.
- gsmp: generic switch management protocol. A porta física é controlada pelo gsmp ao invés dos protocolos do ATM Forum.
- void: UNI pública sem sinalização e ILMI. Um túnel VP pode ser criado em tais portas.

O tipo de acesso ATM só pode ser modificado quando a porta estiver em estado “disabled”. O acesso UNI é o único tipo de acesso válido para as portas virtuais que conectam os módulos MSS-Server, ATM-lan-bridge, ATM-man ou ATM-vídeo. A Tabela I-6 mostra os valores de retorno para essa variável.

Tipo de acesso ATM	Retorno
unknown	1
privateUni	2
privateNetwork	3
publicUni	4
gsmp	5
Void	7
Auto	8
Snoop	9

**Tabela I-6 - Valores de retorno para Tipo de acesso ATM**