

UNIVERSIDADE ESTADUAL DE CAMPINAS
FACULDADE DE ENGENHARIA ELÉTRICA
DEPARTAMENTO DE TELEMÁTICA

**Suporte ao Serviço Não Orientado à Conexão na
Rede Digital de Serviços Integrados de Faixa Larga**

Pedro Graef Junior

Banca examinadora:

**Dr. Ivanil Sebastião Bonatti
Dr. Luís Fernando Brito Baptistella
Dr. Walter da Cunha Borelli**

Suplente:

Dr. Akebo Iamakami

Dissertação apresentada à Faculdade de Engenharia Elétrica
da Unicamp, como requisito parcial para obtenção do título
de Mestre em Engenharia Elétrica.

Maio 1994

Este exemplar corresponde à versão final da tese defendida por <u>Pedro Graef Jr</u>
Julgadora em <u>18 05 94</u> pela Comissão
<u>Bonatti</u> Coordenador

Curriculum Vitae da Banca Examinadora

- **Prof. Dr. Ivanil Sebastião Bonatti** – Presidente
Doutor em Automática – 1981 – Toulouse – França
Título da Tese: Gestion de Réseaux de Service – Application au Réseau Téléphonique Interurbain.
Local de Trabalho: DT – FEE – UNICAMP

- **Prof. Dr. Luís Fernando Brito Baptistella** – Membro
Doutor em Engenharia Elétrica – 1980 – Toulouse – França
Título da Tese: Contribution à l' Optimization Multicritère des Systèmes Dynamiques.
Local de Trabalho: CPqD – TELEBRÁS

- **Prof. Dr. Walter da Cunha Borelli** – Membro
Doutor em Engenharia Elétrica – 1983 – Canterbury – Inglaterra
Título da Tese: Convolutional Codes for Multilevel Data Transmission.
Local de Trabalho: DT – FEE – UNICAMP

- **Prof. Dr. Akebo Iamakami** – Suplente
Doutor em Engenharia Elétrica – 1985 – FEE – UNICAMP
Título da Tese: Contribuição ao Estudo de Sistemas Dinâmicos Interconectados: Estabilidade e Otimização.
Local de Trabalho: DT – FEE – UNICAMP

Este trabalho é dedicado à minha esposa e irmãs pelo incentivo e compreensão e aos meus pais pela priorização da qualidade de nossa educação e formação.

Agradecimentos

Gostaria de manifestar a minha gratidão às pessoas e às instituições que, direta ou indiretamente, contribuíram para a realização deste trabalho:

- ao Prof. Dr. Ivanil Sebastião Bonatti, pela orientação, confiança e desempenho constante dispensados no desenvolvimento deste trabalho.
- aos colegas do grupo de estudos em ATM, Klemz, Froes, Nelson e Armando do Centro de Pesquisa da Telebrás, pelas discussões técnicas e filosóficas sobre tópicos específicos.
- à FAPESP, que através do projeto temático “Redes Metropolitanas Multiserviços” colocou à disposição do Departamento de Telemática recursos de hardware e software que viabilizaram a realização deste trabalho.

Resumo

Este trabalho aborda o suporte ao serviço não orientado à conexão pela Rede Digital de Serviços Integrados de Faixa Larga (B-ISDN), concentrando-se, especificamente, na interconexão de redes locais.

São apresentadas as alternativas de arquitetura de rede, identificando os elementos de rede envolvidos e as opções de implementação destes elementos.

A partir de uma arquitetura de rede e de uma opção para implementação dos elementos de rede, consideradas como as alternativas mais genéricas, é apresentado um detalhamento destes elementos. São descritos os protocolos envolvidos e suas funções, as especificações funcionais na linguagem SDL (Specification and Description Language), assim como as metodologias de desenvolvimento e simulação utilizadas.

Abstract

This work approaches the support of Connectionless Service by the Broadband Integrated Services Digital Network (B-ISDN), concentrating in Local Area Networks interconnection.

The network architecture are presented, identifying the network elements involved and their implementation options.

Selecting one network architecture and one implementation option for the network elements, considered as the more generic alternatives, the details of such elements are shown. The used protocols and their functions are described, as well as, their functional specification using SDL (Specification and Description Language) and the development and simulation methodologies employed.

Índice

Capítulo 1	1
1.1 Introdução	1
1.2 Organização do trabalho	2
1.3 Motivação	2
Capítulo 2 Características básicas da B-ISDN	4
2.1 Técnica de transporte ATM	4
2.2 Modelo de Referência	6
2.3 Tipos de conexões	9
2.4 Rede Virtual de Gerência	10
2.5 Conclusão	11
Capítulo 3 Suporte ao serviço não orientado a conexão	12
3.1 Cenários para o suporte do serviço não orientado à conexão	12
3.1.1 Cenário Indireto	12
3.1.2 Cenário Direto	14
3.1.3 Cenário Híbrido	15
3.2 Aspectos da rede virtual não orientada à conexão	16
3.3 Elementos de rede	18
3.3.1 Elemento Adaptador de Protocolos (IWU)	18
3.3.1.1 "Bridge"	18
3.3.1.2 "Router"	19
3.3.2 Elemento Servidor (CLS)	20
3.3.2.1 Modos de operação do Servidor	20
3.4 Conclusão	21
Capítulo 4 Funcionalidades dos elementos de rede	22
4.1 Camada de Adaptação AAL 4	22
4.1.1 Funções	23
4.1.2 Funções da subcamada SAR	23
4.1.2.1 Estrutura da SAR_PDU	24
4.1.3 Funções da subcamada CPCS	25
4.1.3.1 Estrutura da CPCS_PDU	26
4.1.4 Funções da subcamada SSCS	27
4.1.5 Primitivas associadas à camada AAL 4	27
4.1.5.1 Primitivas de interação com a camada CLNAP	27
4.1.5.2 Primitivas de interação com a camada ATM	27
4.2 Camada CLNAP	29
4.2.1 Funções da camada CLNAP	29
4.2.1.1 Estrutura da CLNAP_PDU	32
4.2.2 Primitivas associadas à camada CLNAP	33
4.2.2.1 Primitivas de interação com a camada IP	34
4.3 Camada IP	35

4.3.1 Funções da camada IP	35
4.3.1.1 Estrutura da IP_PDU	40
4.3.2 Primitivas associadas à camada IP	42
4.3.2.1 Primitivas de interação com a camada LLC	42
4.4 Conclusão	42
Capítulo 5 Especificação e simulação dos elementos de rede	44
5.1 Especificação formal	44
5.2 Especificação funcional da B-ISDN	45
5.3 Especificação funcional do IWU	49
5.3.1 Descrição dos tipos	50
5.3.2 Descrição dos blocos	53
5.3.2.1 Descrição dos processos	55
5.3.3 Configuração para simulação	56
5.3.3.1 Procedimento	59
5.4 Especificação funcional do CLS	60
5.4.1 Descrição dos tipos	61
5.4.2 Descrição dos blocos	61
5.4.2.1 Descrição dos processos	63
5.4.3 Configuração para simulação	63
5.4.3.1 Procedimento	64
5.5 Estudo de caso	66
5.6 Conclusão	70
Capítulo 6 Conclusão	72
Apêndice A Modelo em camadas	74
A.1 Conceitos básicos	74
A.2 Primitivas de serviço	76
A.3 Protocolo	77
A.4 Unidades de dado	77
Apêndice B Características básicas da linguagem SDL	78
B.1 Introdução	78
B.2 Conceitos básicos	78
B.2.1 Estruturação e partição do sistema	79
B.2.1.1 Subestruturas	80
B.2.2 Descrição dos elementos do sistema	81
B.2.2.1 Descrição do sistema	81
B.2.2.2 Descrição do bloco	83
B.2.2.3 Descrição do processo	85
B.2.2.4 Descrição do serviço	89
B.2.2.5 Descrição de procedimento e macro	90
B.2.3 Descrição de dados	92
B.2.3.1 Tipos pré-definidos	92
B.2.3.2 Tipos estruturados pré-definidos	92
B.2.3.3 Operadores pré-definidos	93
B.2.3.4 Definição de novos tipos	93
B.2.4 Comunicação entre processos	94
B.2.5 Descrição de tempo	96
B.2.6 Construções da linguagem SDL	98
B.3 Restrições da linguagem SDL	99
Apêndice C Ferramenta SDT	102

C.1	Resumo dos principais aplicativos	102
C.2	Procedimento utilizado	104
C.3	Compatibilidade com SDL e MSC	105
C.4	Restrições da ferramenta SDT-2	105
C.5	Ambiente utilizado	106
Apêndice D Glossário		107

Figuras

Figura 1.1: Serviços e aplicações nas redes experimentais de faixa larga [11] . . .	3
Figura 2.1: Relação entre VPI e VCI	4
Figura 2.2: Célula ATM	5
Figura 2.3: Conexão entre usuários na rede ATM	6
Figura 2.4: Modelo de referência dos protocolos da B-ISDN	7
Figura 2.5: Processo de estabelecimento de uma chamada na B-ISDN	8
Figura 2.6: Rede Virtual de Gerência	10
Figura 3.1: Cenário indireto (caso 1)	13
Figura 3.2: Cenário indireto (caso 2)	13
Figura 3.3: Cenário direto	14
Figura 3.4: Rede virtual não orientada à conexão	15
Figura 3.5: Perfil de tráfego em uma rede local	16
Figura 3.6: IWU implementado como "Bridge"	18
Figura 3.7: IWU implementado como "Router"	19
Figura 3.8: Estrutura de protocolos do servidor	20
Figura 3.9: Primeira célula da unidade de dado CLNAP	21
Figura 4.1: Organização da camada AAL 4	23
Figura 4.2: Estrutura da SAR_PDU	24
Figura 4.3: Processos de segmentação e remontagem	25
Figura 4.4: Estrutura da CPCS_PDU	26
Figura 4.5: Interfaces com a camada CLNAP	29
Figura 4.6: Estrutura da CLNAP_PDU	32
Figura 4.7: Interfaces com a camada CLNAP	35
Figura 4.8: Processos de remontagem e fragmentação de unidades IP_PDU, considerando unidades de tamanhos arbitrários	36
Figura 4.9: Mapeamento de endereços individuais. Note que para a rede local "A" existe uma correspondência com as tabelas 4.8 e 4.9	38
Figura 4.10: Formação de um endereço de grupo	39
Figura 4.11: Estrutura da IP_PDU	40
Figura 5.1: Cenário da B-ISDN adotado	45
Figura 5.2: Definição do sistema B-ISDN	46
Figura 5.3: Definição do bloco "Rede_virtual"	47
Figura 5.4: Definição do sistema "IWU"	49
Figura 5.5: IP_PDU_type	50
Figura 5.6: ADDRESS_type	50
Figura 5.7: TOS_type	51
Figura 5.8: SAR_PDU_type	51
Figura 5.9: Definição do bloco "IWU_TX_Handler"	53
Figura 5.10: Definição do bloco "IWU_RX_Handler"	54
Figura 5.11: Modelo de rede para simulação do sistema "IWU"	57
Figura 5.12: Definição do sistema "CLS"	60
Figura 5.13: SAR_PDU_type	61
Figura 5.14: Definição do bloco "Server"	62
Figura 5.15: Modelo de rede para simulação do sistema "CLS"	64

Figura 5.16: Parâmetros “SAR_PDU” e “CI_in” definidos	67
Figura 5.17: Parâmetros “SAR_PDU” e “CI_out” observados	69
Figura 5.18: Parte do “trace” de simulação	70
Figura A.1: Relação entre entidades, conexão, SAP e CE	75
Figura A.2: Associação entre conexões AAL e ATM	75
Figura A.3: Primitivas de serviço	76
Figura A.4: Relação entre as unidades de dado	77
Figura B.1: Hierarquia de descrição do sistema	80
Figura B.2: Subestrutura de canal	81
Figura B.3: Subestrutura de bloco	81
Figura B.4: Sistema “CLS”	82
Figura B.5: Bloco “Server”	84
Figura B.6: Processo “Message_Dmux”	86
Figura B.7: Processo “Message_Dmux” particionado em serviços	88
Figura B.8: Descrição do serviço “Init”	89
Figura B.9: Definição do procedimento “Find_router”	90
Figura B.10: Definição da macro “init_Inst_table”	91
Figura B.11: Dinâmica da comunicação entre processos	95
Figura B.12: Manipulação dos tipos <i>time</i> e <i>duration</i>	96
Figura B.13: Definição e controle do temporizador	97
Figura B.14: Construções da linguagem SDL	98
Figura B.15: Construções da linguagem SDL (continuação)	99
Figura B.16: Interfaces do processo “IP_tx_Dmux”	100
Figura B.17: Definição de prioridade de um sinal	100
Figura B.18: Exemplo de definição do operador “MOD”	101
Figura C.1: Aplicativos e bibliotecas da ferramenta SDT	103
Figura C.2: Procedimento de desenvolvimento	105

Tabelas

Tabela 2.1: Tipos de AAL por classe de serviço	5
Tabela 4.1: Identificação da SAR_PDU	24
Tabela 4.2: Valores possíveis para o campo LI	25
Tabela 4.3: Opções de conexões AAL por endereço de destino	30
Tabela 4.4: Opções de conexões AAL por classe de qualidade de serviço	30
Tabela 4.5: Decomposição do endereço de grupo	31
Tabela 4.6: Restrição de encaminhamento	31
Tabela 4.7: Estrutura dos campos DA e SA	33
Tabela 4.8: Associação entre os endereços IP e MAC de uma rede local	37
Tabela 4.9: Associação dos endereços IP, por rede local, com os endereços de rede E.164 correspondentes	37
Tabela 4.10: Associação entre os endereços de grupo IP, MAC e E.164	37
Tabela 4.11: Manipulação dos campos OFF e FLA no processo de fragmentação	41
Tabela 5.1: Correlação entre os canais dos sistemas "IWU" e "B-ISDN"	50
Tabela 5.2: AAL_table_type (configuração inicial)	51
Tabela 5.3: Associação entre as conexões AAL e ATM ("CI_TABLE_type") ..	57
Tabela 5.4: Opções de roteamento ("Route_table_type")	57
Tabela 5.5: Associação entre os endereços E.164 e IP por rede local ("ADDI_tx_type")	58
Tabela 5.6: Associação entre os endereços IP e MAC das redes locais ("ADDI_rx_type")	58
Tabela 5.7: Definição dos endereços de grupo ("ADDG_type")	58
Tabela 5.8: Opções de roteamento ("Route_table_type")	64

Capítulo 1

1.1 Introdução

O desenvolvimento tecnológico associado à automação dos processos industriais e a informatização da sociedade vêm provocando alterações no perfil da demanda dos serviços de comunicações, assim como na concepção das redes de telecomunicações.

No ambiente do usuário ocorre uma proliferação de microcomputadores e estações de trabalho com alta capacidade de processamento e novas aplicações, formando redes locais.

Na rede de telecomunicações ocorre a disseminação de meios de transmissão com elevada capacidade de transferência de informação e alta confiabilidade, assim como a concepção de comutadores com protocolos de comunicação de alto desempenho e flexíveis, de forma a integrar os diferentes tipos de serviços.

Com o objetivo de atender a demanda de interconexão de redes locais e centros computacionais geograficamente distribuídos, diferentes soluções de rede, consideradas de “faixa larga”¹, surgiram, tais como:

- Rede “Cross-Connect” oferecendo circuitos dedicados de 2.0 a 34 Mbps (Mega bits por segundo) ou de 1.5 a 45 Mbps, provisionados pelas redes de transporte plesiócronicas ([1] e [2]) e síncrona ([3] e [4]).
- Rede “Frame Relay” [5], que corresponde a uma evolução das redes de pacotes baseadas no protocolo X.25 [6] e que oferece conexões virtuais semipermanentes a 2 Mbps.
- Rede Metropolitana (MAN – Metropolitan Area Network), suportando a integração dos serviços de vídeo, voz e dados, mas concebida principalmente para a aplicação de dados através do serviço SMDS (Switched Multi-Megabit Data Service) [7]. Esta oferece ao usuário acessos com taxas de 2.0 a 34 Mbps e de 1.5 a 45 Mbps.
- Rede Digital de Serviços Integrados de Faixa Larga (B-ISDN – Broadband Integrated Service Digital Network), preconizada pelo CCITT² (International Telegraph and Telephone Consultative Committee) para suportar a demanda dos serviços atuais e futuros. Esta se baseia na técnica de multiplexação e comutação ATM (Asynchronous Transfer Mode) [8], que corresponde a uma evolução da rede “Frame Relay”. A B-ISDN suporta a integração dos serviços de vídeo, voz e dados e oferece ao usuário acessos com as taxas máximas de 155.52 Mbps e 622.08 Mbps.

1. Tradicionalmente uma rede é considerada de “faixa larga” quando oferece suporte para taxas de acesso superiores a 2 Mbps.

2. Atualmente ITU-TSS (International Telecommunication Union – Telecommunication Standardization Sector).

1.2 Organização do trabalho

Este trabalho apresenta alguns aspectos relativos à implementação dos elementos de rede envolvidos no suporte do serviço não orientado à conexão na Rede Digital de Serviços Integrados de Faixa Larga (B-ISDN), concentrando-se, especificamente, na interconexão de redes locais.

O trabalho está dividido em seis capítulos, dos quais o capítulo 1, introdutório, apresenta o contexto tecnológico e mercadológico que motivaram o desenvolvimento deste tema. O capítulo 2 apresenta conceitos e princípios da B-ISDN que servem de subsídios à exposição dos capítulos posteriores.

O capítulo 3 aborda alguns aspectos e opções de implementação, no nível de arquitetura de rede e no nível de elemento de rede, envolvidos no suporte ao serviço não orientado à conexão.

No capítulo 4 é apresentado um detalhamento dos elementos de rede descrevendo os protocolos envolvidos e suas funções, considerando um cenário selecionado com base nas características das opções expostas no capítulo 3.

No capítulo 5 são apresentados os modelos e especificações funcionais dos elementos de rede na linguagem SDL (Specification and Description Language) [9] e as metodologias de desenvolvimento e simulação utilizadas.

O último capítulo conclui o trabalho relacionando as etapas e estudos complementares, identificados como necessários, para a transformação das especificações funcionais dos elementos de rede em produtos.

Quatro apêndices complementam este trabalho, apresentando:

- Apêndice A: os conceitos básicos e terminologias do modelo em camadas e de protocolos para sistemas abertos.
- Apêndice B: os conceitos básicos da linguagem SDL e os procedimentos de particionamento do sistema em diferentes níveis hierárquicos de abstração.
- Apêndice C: os recursos da ferramenta SDT (SDL Design Tool) [10] utilizados no desenvolvimento deste trabalho.
- Apêndice D: o glossário.

1.3 Motivação

Basicamente três aspectos serviram como agentes motivadores ao desenvolvimento deste trabalho. O primeiro aspecto, mercadológico, decorre do fato de que a principal demanda atual para a B-ISDN, conforme estudo realizado em [11], é a transferência de dados a altas taxas, especialmente na interconexão de redes locais e metropolitanas.

A figura 1.1 mostra esta tendência, apresentando o grau de incidência dos diversos serviços e aplicações nas experiências de redes faixa larga em nível mundial.

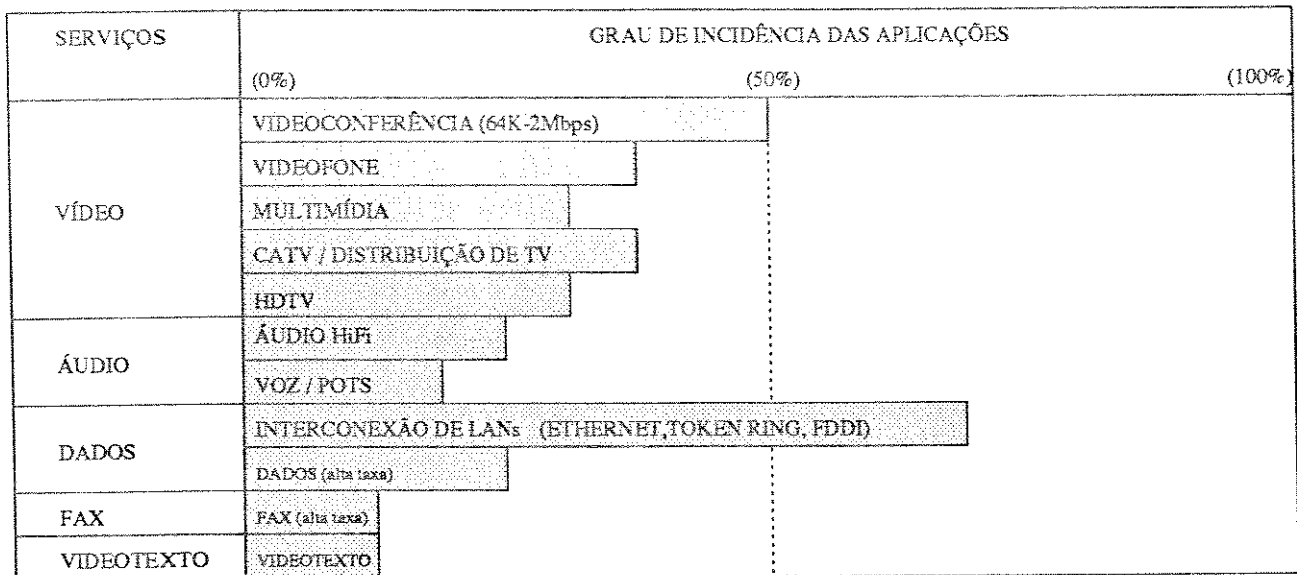


Figura 1.1: Serviços e aplicações nas redes experimentais de faixa larga [11]

O segundo aspecto está no desafio levantado pelo paradoxo de que a B-ISDN integra as diferentes aplicações (dados, áudio e vídeo) do usuário através da técnica de transporte ATM orientada à conexão e que as redes locais e metropolitanas operam essencialmente com protocolos não orientados à conexão.

O terceiro aspecto motivador está no aproveitamento dos resultados deste trabalho nas atividades de adição de facilidades de faixa larga ao equipamento Trópico [12] (Central Telefônica com programa armazenado) em desenvolvimento no CPqD da Telebrás. Esta atividade é atualmente realizada no âmbito da Pesquisa Aplicada em Sistemas de Comutação de Alto Desempenho (PA SCAD).

Capítulo 2

Características básicas da B-ISDN

2.1 Técnica de transporte ATM

A técnica de comutação e multiplexação ATM (Asynchronous Transfer Mode) [8] foi adotada para a B-ISDN por ser flexível e eficiente, de forma a suportar os diferentes requisitos dos diversos serviços e aplicações do usuário.

A técnica ATM está baseada em um sistema de transporte de unidades de dados de tamanho fixo, denominadas por células ATM.

As células ATM são formadas por um campo para transporte de informação de usuário e um cabeçalho cuja principal função é o encaminhamento da célula através da rede. Este encaminhamento é definido por dois identificadores:

- VPI (Virtual Path Identifier): identificador do caminho lógico.
- VCI (Virtual Channel Identifier): identificador do canal lógico de um caminho lógico.

A figura 2.1 ilustra a relação entre os identificadores VPI e VCI.

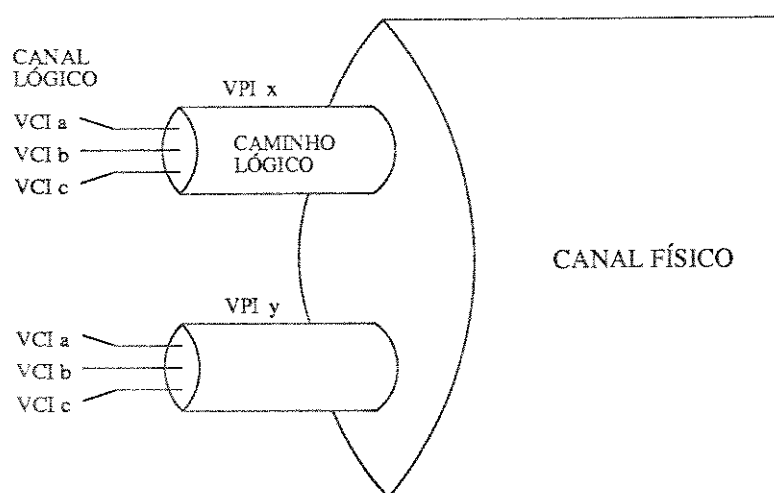


Figura 2.1: Relação entre VPI e VCI

A técnica ATM é dita orientada à conexão no sentido que a informação do usuário é transportada por células encaminhadas através de conexões virtuais, pré-estabelecidas, que são definidas pelos identificadores VPI e VCI.

A figura 2.2 apresenta a estrutura de uma célula ATM. Uma descrição detalhada da estrutura do cabeçalho da célula pode ser encontrada em [13].

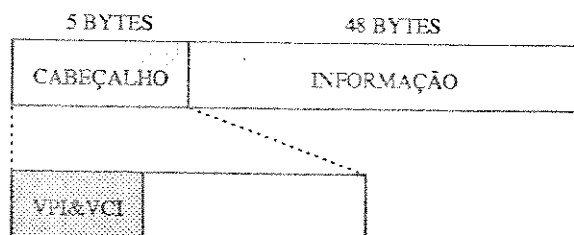


Figura 2.2: Célula ATM

Para que os diversos serviços utilizem o modo de transporte ATM, foram definidas estruturas de adaptação denominadas AAL (ATM Adaptation Layer) [14], com funções específicas para cada classe de serviço da aplicação do usuário, conforme apresentado na tabela 2.1.

Classe de serviço	Tipo de AAL
A	AAL 1
B	AAL 2
C	AAL 3 ou AAL 5 ³
D	AAL 4 ⁴

Tabela 2.1: Tipos de AAL por classe de serviço

Onde,

Classe A: Representa os serviços isócronos, ou seja, com taxa de bits constante e com restrições de tempo na transmissão das células, tais como no transporte de:

- circuitos síncronos da hierarquia de 64 Kbps até 1920 Kbps [15] e [4].
- circuitos assíncronos de 1544 Kbps, 2048 Kbps, até 44.736 Kbps [1].
- sinais de vídeo
- sinais de voz e de áudio de alta qualidade.

Classe B: Representa os serviços com taxa de bits variável e com restrições de tempo na transmissão das células, como no caso do transporte de sinais de vídeo codificados em taxa variável.

3. A AAL 5, definida mais recentemente, é uma versão simplificada da AAL 3, com menor tempo de processamento, porém, com uma menor capacidade de detecção de células inválidas.

4. Atualmente, existem propostas para substituição da AAL 4 pela AAL 5.

Classe C: Representa os serviços assíncronos, sem restrições de tempo na transmissão das células e transporte orientado à conexão, ou seja, definido por três fases: estabelecimento da conexão entre os usuários de origem e destino, transferência das unidades de dado de informação pela conexão definida e a fase de liberação da conexão. Note que uma conexão, entre usuários, é caracterizada por um conjunto de conexões ATM ao longo da rede, conforme ilustrado na figura 2.3.

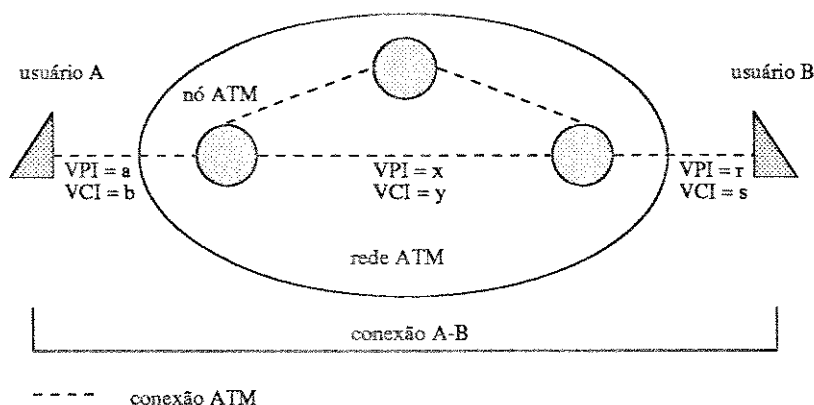


Figura 2.3: Conexão entre usuários na rede ATM

Nesta classe estão, por exemplo, os serviços X.25, "Frame Relay" e o próprio serviço de estabelecimento de chamada da B-ISDN [16].

Classe D: Representa os serviços assíncronos, sem restrições de tempo na transmissão das células e transporte não orientado à conexão. Esta classe de serviço caracteriza-se pelo transporte das unidades de dado através da rede, sem o estabelecimento de uma conexão entre os usuários de origem e destino. O encaminhamento das unidades de dado é realizado em função do endereço do usuário de destino, contido no campo de informação da célula. Nesta classe estão, por exemplo, os protocolos de transporte das redes locais de computadores como, por exemplo, o TCP/IP (Transmission Control Protocol/ Internet Protocol) [17].

2.2 Modelo de Referência

O modelo de referência para os protocolos envolvidos na B-ISDN, considerado neste trabalho, está baseado nas recomendações [18] e [19] do CCITT e usa os princípios e conceitos de estruturação em camadas similares aos definidos pelo modelo OSI (Open Systems Interconnection) [20] da ISO (International Standard Organization).

Para a representação da B-ISDN o modelo de referência adota planos separados para os protocolos e funções de usuário, controle e gerenciamento, conforme apresentados na figura 2.4.

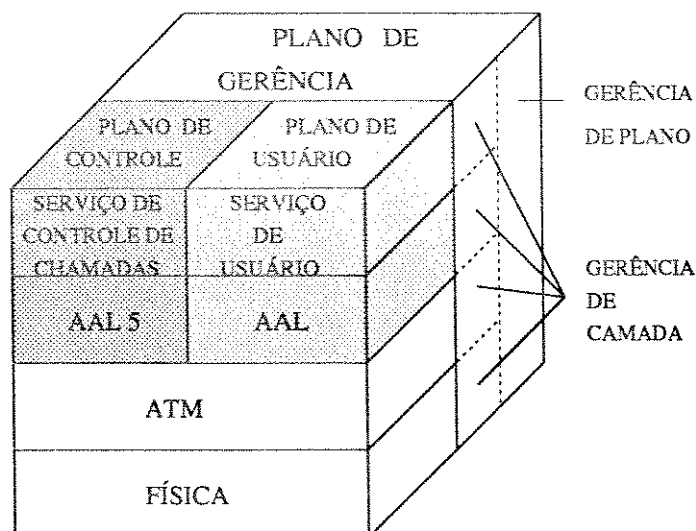


Figura 2.4: Modelo de referência dos protocolos da B-ISDN

O plano de controle, com sua estrutura em camadas, realiza as funções de controle de uma chamada e das conexões envolvidas⁵, executando os procedimentos de sinalização necessários para o estabelecimento, liberação, negociação e renegociação de chamadas e conexões, e a alocação de recursos na rede.

O plano de usuário, também estruturado em camadas, é responsável pelo transporte de informação entre usuários, através de uma ou mais conexões estabelecidas pelo plano de controle.

O plano de gerência provê as funções de administração, supervisão, operação, manutenção do elemento de rede e a troca de informações entre o plano de controle e o plano de usuário. Além disto, este plano é responsável pela monitoração das conexões estabelecidas e interface com o centro de gerência da rede.

O plano de gerência é formado pelas gerências de camada e pela gerência de plano. As gerências de camada realizam as funções de gerenciamento locais, específicas de cada camada, e a troca de informações entre as camadas do plano de usuário e a gerência de plano.

A gerência de plano executa as funções de gerenciamento do elemento de rede como um todo, coordenando a interação entre os planos de controle, de usuário e entre as gerências de camada.

As camadas FÍSICA e ATM são comuns aos planos de controle e de usuário, multiplexando as informações provenientes destes planos. A camada FÍSICA realiza as funções de transmissão, tais como geração e recuperação de relógio e sincronismo de quadro, delineamento da célula ATM, geração e verificação do CRC (Cyclic Redundancy Check) referente ao cabeçalho da célula ATM e embaralhamento e desembaralhamento da célula.

5. Uma chamada multimídia pode, por exemplo, envolver diferentes conexões para o transporte de informações de áudio, vídeo e dados.

A camada ATM [13] basicamente executa as funções de multiplexação e demultiplexação de células, controle de fluxo, descarte seletivo de células e detecção de células inválidas.

A camada AAL [21] realiza as funções necessárias à compatibilização do serviço do usuário com o serviço prestado pela camada ATM. A camada AAL executa as funções de segmentação da unidade do usuário em células, remontagem de células e detecção de células inválidas ou fora de sequência. Além disto, esta camada conforme o serviço do usuário executa funções, tais como:

- Geração e recuperação de relógio e sincronismo de quadro para serviços classe A (AAL 1) e B (AAL 2).
- Funções de rede [22] como, por exemplo, o controle de fluxo e a recuperação de erros para os serviços classe C (AAL 3 ou AAL 5).

Para o plano de controle, o CCITT especificou [23] a camada de Adaptação AAL 5, que implementa um subconjunto das funções executadas pela camada de Adaptação AAL 3.

A figura 2.5 apresenta o fluxo de interações entre os planos, considerados no modelo de referência, durante a fase de estabelecimento de uma chamada e conexões associadas ao plano de usuário.

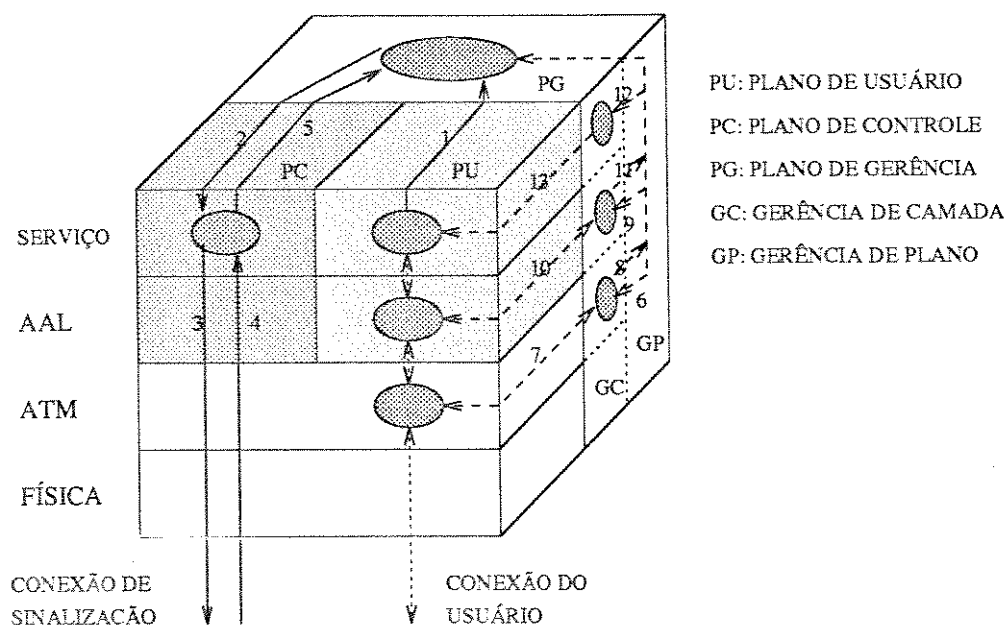


Figura 2.5: Processo de estabelecimento de uma chamada na B-ISDN

Fluxos 1 e 2: O usuário inicia a chamada através de uma requisição ao plano de controle que acontece por intermédio do plano de gerência. Nesta requisição o usuário define um conjunto de parâmetros que caracterizam a chamada, tais como: as classes de serviço e número de conexões envolvido, perfil de tráfego, tempo de atraso máximo e taxa máxima de perda de células.

- Fluxos 3 e 4:** Representam os processos de verificação de compatibilidade entre as entidades, de origem e destino, do plano de controle e de disponibilidade de recursos na rede, compatíveis com os parâmetros definidos para a chamada. Esta verificação é realizada através de uma conexão dedicada à sinalização e estabelecida previamente por procedimentos de gerência de rede através do plano de gerência.
- Fluxo 5:** Indica ao plano de gerência do resultado da negociação de parâmetros realizados através dos fluxos 3 e 4. Considera-se, neste exemplo, que a chamada foi realizada com sucesso.
- Fluxos 6 e 9:** Representam, respectivamente, a passagem de informações de configuração relativas, às camadas ATM e AAL, necessárias ao estabelecimento da chamada.
- Fluxos 7 e 10:** Representam, respectivamente, o processo de interação entre as entidades das camadas ATM e AAL com as entidades das gerências de camada correspondentes. Neste processo são realizadas as configurações dos serviços prestados por estas camadas, assim como a associação dos identificadores de conexão:
- da camada AAL com ATM (fluxo 10).
 - da camada ATM com FÍSICA (fluxo 7).
- Fluxos 8 e 11:** Indicam a confirmação, ao plano de gerência, da execução das configurações e do estabelecimento das associações entre os identificadores de conexões iniciados, respectivamente, pelos fluxos 7 e 10.
- Fluxos 12 e 13:** Indicam o término do estabelecimento da chamada e passagem dos identificadores de conexão AAL, pelos quais os dados do usuário serão transportados.

2.3 Tipos de conexões

O usuário pode utilizar conexões sob demanda ou semipermanentes para o transporte de sua informação.

Conexões sob demanda são aquelas controladas, estabelecidas e liberadas diretamente por ações do usuário, através do plano de controle, conforme representado na figura 2.5. Este tipo de conexão é utilizado, por exemplo, no serviço telefônico.

Conexões semipermanentes são aquelas cujo controle, estabelecimento e liberação são realizados por procedimentos de gerência de rede, através do plano de gerência. Neste caso os parâmetros desejados para a conexão são negociados entre o usuário e o operador da rede na ocasião de sua subscrição. Este conceito de conexão aplica-se, por exemplo, às linhas privativas.

No transporte de dados não orientados à conexão, onde as unidades de dados são independentes, a utilização de conexão sob demanda implicaria em procedimentos de estabelecimento e liberação para cada mensagem do usuário. Considerando que o tempo médio entre mensagens em uma rede local (da ordem de mili segundos) é muito menor que os tempos de estabelecimento e liberação de conexões (da ordem de segundos), o uso de conexões sob

demanda inviabilizaria a implementação dos equipamentos de rede. Desta forma, o transporte de dados não orientados à conexão é realizado através de uma rede virtual de conexões semipermanentes, de modo que o plano de controle nos elementos de rede envolvidos neste serviço não é implementado.

2.4 Rede Virtual de Gerência

Neste trabalho é considerado que os elementos que compõem a B-ISDN são supervisionados, configurados e reconfigurados pela gerência de rede por intermédio de uma rede virtual de gerência.

Esta rede é definida por um conjunto de conexões semipermanentes, dedicadas ao transporte de informações de gerência, através das quais o centro de gerência tem acesso ao plano de gerência dos elementos de rede conforme representado na figura 2.6.

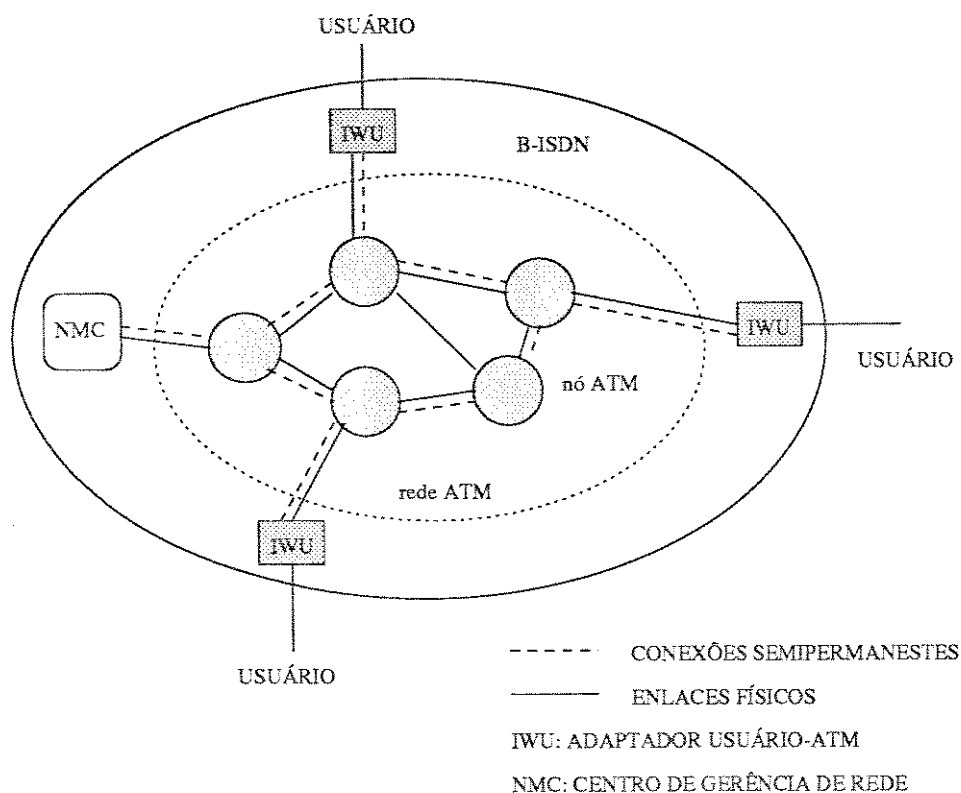


Figura 2.6: Rede Virtual de Gerência

Os mecanismos e procedimentos de atuação da gerência de rede não fazem parte do escopo deste trabalho, que considera as estruturas de dados de configuração dos elementos de rede estáticas e pré-definidas, enfatizando, desta forma, as funcionalidades e protocolos referentes ao plano de usuário.

2.5 Conclusão

Este capítulo abordou as características básicas da B-ISDN, apresentando os conceitos e definições associados à técnica de transporte ATM, ao modelo de referência utilizado na B-ISDN, aos tipos de conexões e à rede virtual de gerência.

Além disto, este capítulo insere o conceito do serviço não orientado à conexão no contexto da B-ISDN e justifica a utilização de conexões semipermanentes no provisionamento deste serviço.

Capítulo 3

Suporte ao serviço não orientado a conexão

Para prover o serviço não orientado à conexão na B-ISDN o CCITT especificou o protocolo CLNAP^{6, 7} (Connectionless Network Access Protocol) [25] que atua, segundo o modelo de referência, em uma camada de nível superior a AAL 4, no plano de usuário.

A principal função executada pela camada CLNAP é o roteamento das unidades de dado do usuário.

3.1 Cenários para o suporte do serviço não orientado à conexão

Três cenários são previstos pelo CCITT para o suporte do serviço não orientado à conexão na B-ISDN [26] e [27].

3.1.1 Cenário Indireto

No cenário indireto a B-ISDN oferece apenas a infraestrutura de conexões ATM semipermanentes, transparente ao serviço demandado, para o transporte das unidades de dados do usuário. Neste cenário duas topologias de rede são possíveis:

- 1 – Os elementos adaptadores de protocolo IWU (Interworking Unit) são interconectados diretamente, ficando assim responsáveis pela execução das funcionalidades da camada CLNAP, inclusive o roteamento das unidades de dados. A figura 3.1 exemplifica esta topologia.

Esta topologia possui dois inconvenientes que se agravam à medida que o número “N” de elementos IWU, que deseja se comunicar, aumenta. O primeiro é que a rede deve manter e tratar “ $N*(N-1)/2$ ” conexões semipermanentes, e o segundo é que as tabelas de roteamento estão distribuídas nos elementos IWU, tornando complexo o controle de consistência destas.

- 2 – Os elementos IWU são interconectados entre si através de um ou mais elementos servidores CLS (Connectionless Server), externos a B-ISDN, que executam a função de roteamento das unidades de dado (camada CLNAP). A figura 3.2 ilustra esta topologia.

Esta segunda topologia não apresenta os inconvenientes da primeira, porém, centraliza o tráfego das unidades de dado, exigindo elementos servidores implementados com alta capacidade de roteamento.

6. O protocolo CLNAP foi definido de forma a manter uma estreita compatibilidade com o serviço SMDS (Switched Multimegabit Data Service) [7] especificado pelo Bellcore.

7. Atualmente, estudos estão sendo realizados para a definição do protocolo CLNIP (Connectionless Network Interface Protocol) [24] para a interface de rede. Desta forma, este trabalho generalizará o uso do protocolo CLNAP para as interfaces de usuário e de rede.

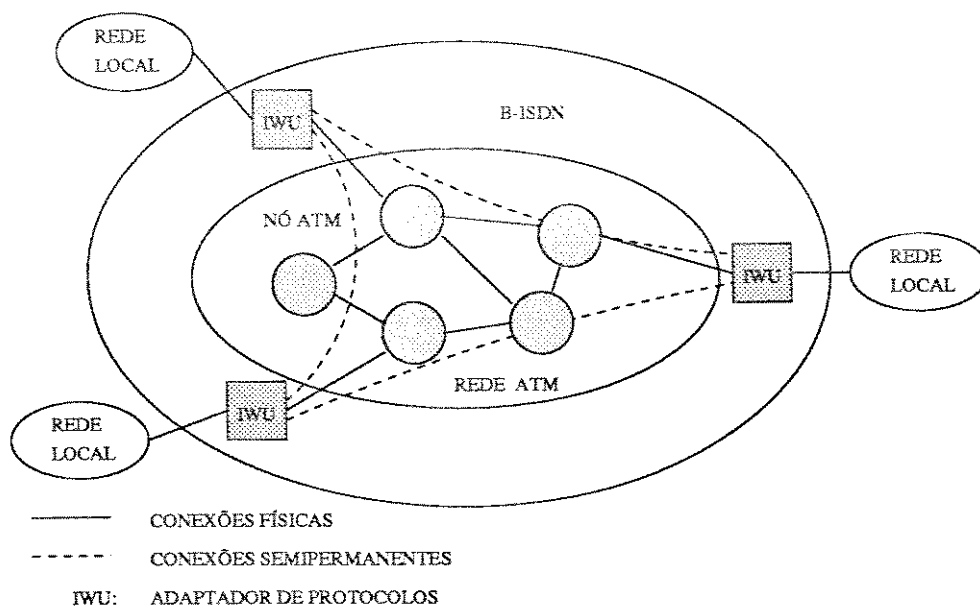


Figura 3.1: Cenário indireto (caso 1)

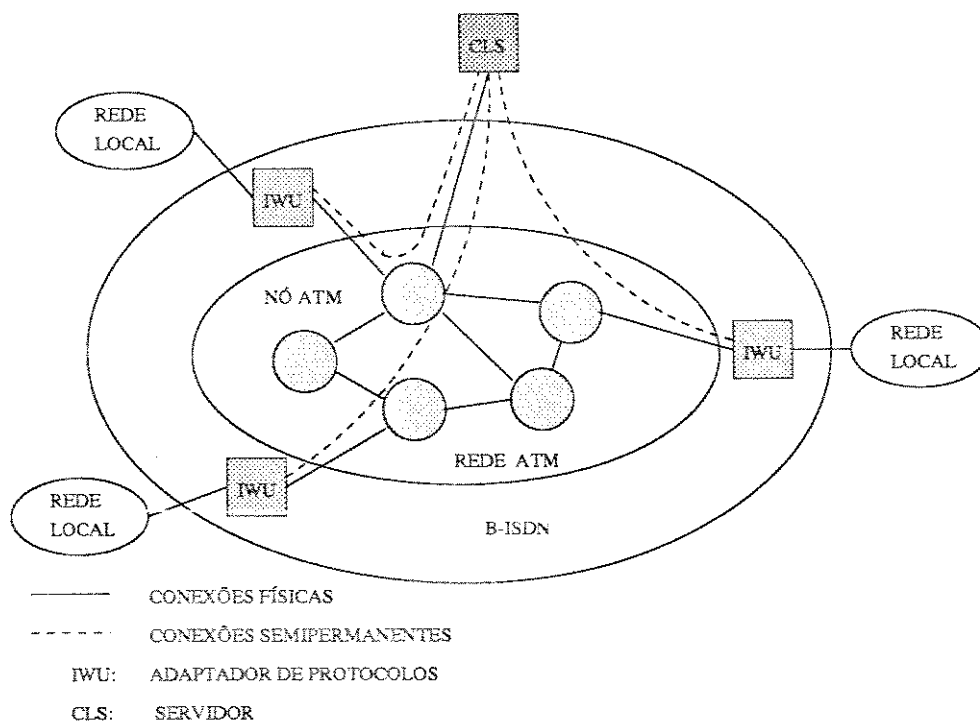


Figura 3.2: Cenário indireto (caso 2)

3.1.2 Cenário Direto

O cenário direto possui a mesma topologia de rede apresentada no segundo caso do cenário indireto, porém, com os elementos servidores (CLS) implementados internamente a B-ISDN. A diferença, neste caso, é que a conexão semipermanente de acesso do elemento IWU à rede ATM fica designada exclusivamente ao serviço não orientado à conexão (classe D).

A figura 3.3 representa este cenário, onde os elementos servidores são implementados adicionando-se as funcionalidades das camadas CLNAP e AAL 4 ao nó ATM.

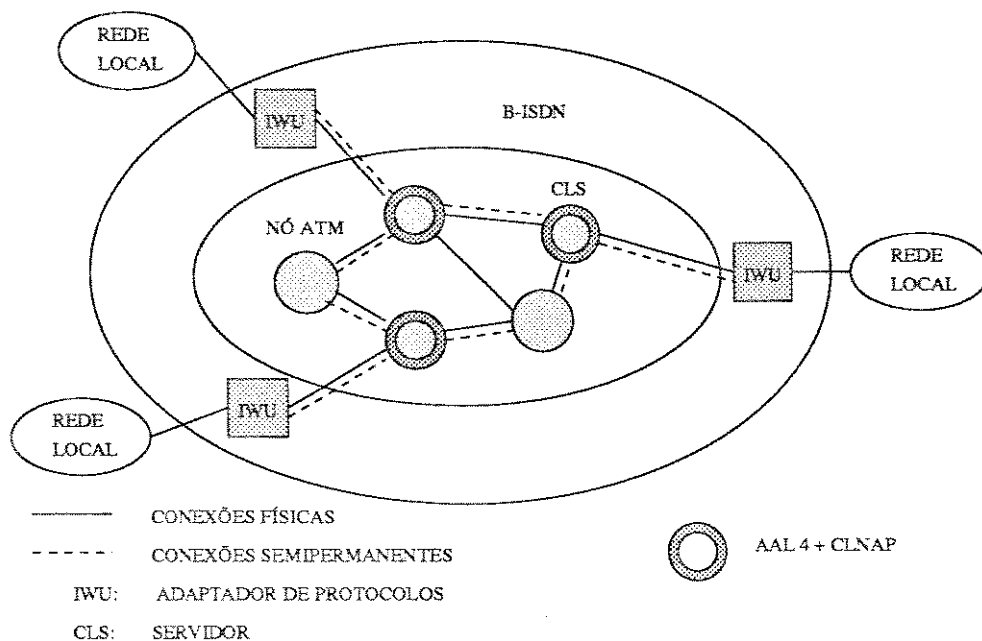


Figura 3.3: Cenário direto

Esta topologia pode ser estratificada em dois planos, onde um representa a rede virtual não orientada à conexão e o outro a rede de transporte ATM. A figura 3.4 ilustra estes planos.

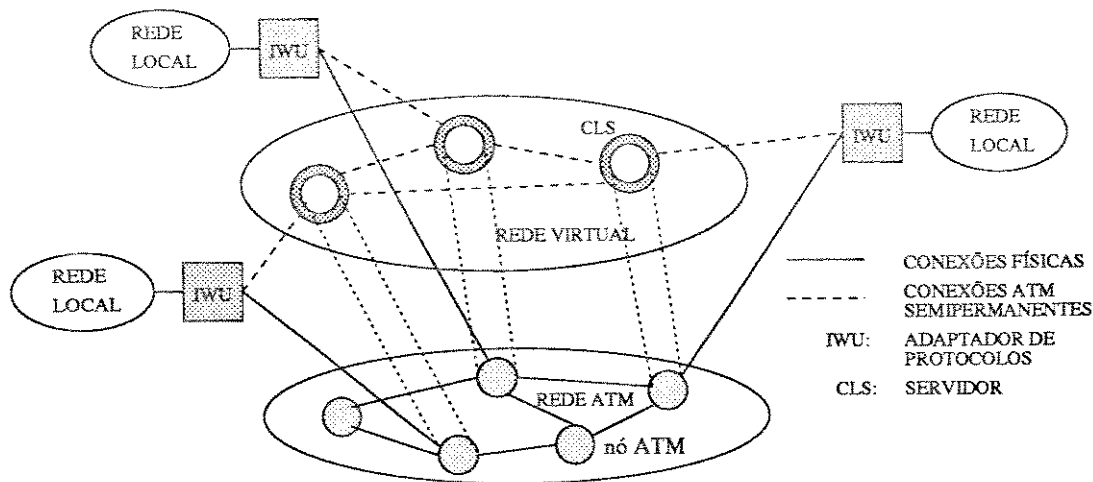


Figura 3.4: Rede virtual não orientada à conexão

O cenário direto possui as seguintes vantagens em relação ao indireto:

- O servidor, pelo qual o elemento IWU tem acesso à rede, pode atuar como um filtro de rede, descartando as células, que compõem a unidade de dado do usuário, por exemplo, no caso de detecção de célula inválida, fora de sequência, com informações de endereçamento de origem e de destino incompatíveis ou que estejam excedendo os limites de tráfego especificados. Esta função do servidor pode evitar a propagação de tráfego desnecessário ao longo da rede.
- Diminuição do tráfego a ser suportado pela rede ATM quando se deseja encaminhar a mesma unidade de dado do usuário para mais de um destino (veja descrição da função de endereçamento de grupo⁸ no capítulo 4). Esta diminuição acontece porque o servidor só replica a unidade de dado em pontos específicos da rede, determinados em função dos endereços dos usuários de destino e da topologia da rede virtual.
- A rede virtual não orientada à conexão fornece intrinsicamente a opção de rotas alternativas, que podem ser devidamente utilizadas em situações de congestionamento, ou de falhas da rede.

3.1.3 Cenário Híbrido

O cenário híbrido corresponde à integração das soluções de suporte Direto e Indireto dentro da B-ISDN, de forma que esta passa a oferecer as potencialidades de cada topologia em função dos requisitos de serviço desejados pelos usuários.

Assim, por exemplo, se existe um elevado interesse de tráfego entre dois usuários, a solução do cenário Indireto (caso 1) pode ser mais adequada. Já, no caso da interconexão de vários usuários, com baixo interesse de tráfego, a solução do cenário Direto torna-se mais adequada.

8. Também conhecida por função "multicast".

3.2 Aspectos da rede virtual não orientada à conexão

A seguir são apresentados, qualitativamente, alguns aspectos que envolvem a rede virtual não orientada à conexão:

- 1- Alocação de Banda das conexões semipermanentes: De um modo geral os usuários que demandam o transporte não orientado à conexão, como é o caso das redes locais, possuem um perfil de tráfego caracterizado por surtos de unidades de dado de intensidade, duração e frequência aleatórios conforme exemplificado na figura 3.5.

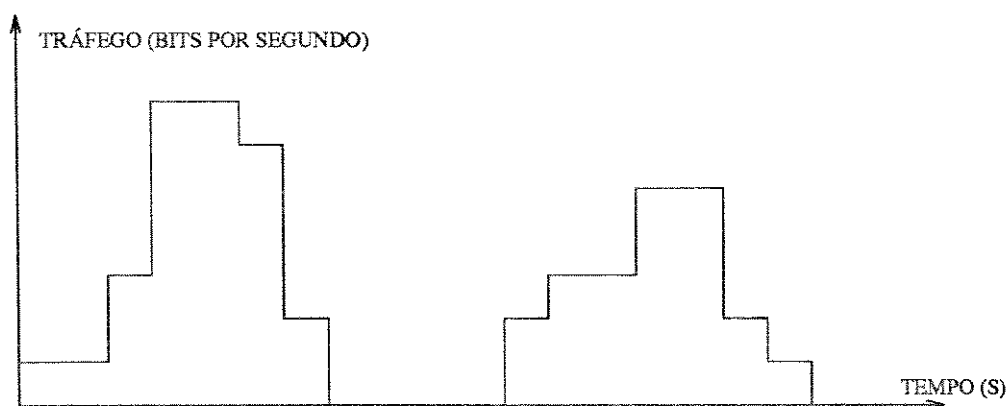


Figura 3.5: Perfil de tráfego em uma rede local

Esta característica de tráfego torna complexo o processo de reserva e alocação de banda⁹ em uma conexão semipermanente.

Três procedimentos para alocação e reserva de banda em uma conexão semipermanentes são, a seguir, brevemente discutidos:

- A. Alocação de banda de pior caso: Este é um método estático que compreende a reserva do valor máximo de banda prevista para a conexão. Este procedimento por ser conservativo ao extremo implica, muitas vezes, em um consumo de recursos desnecessários da rede, de forma que este tende a não ser utilizado na prática, a não ser no caso de um número bastante reduzido de usuários. Este procedimento se torna viável quando os recursos de transmissão são abundantes e baratos quando comparados com os recursos de comutação.
- B. Renegociação de banda a cada surto de unidades de dado: Este procedimento dinâmico consiste [28] na alocação inicial de um valor mínimo de banda para cada conexão semipermanente. Quando o número de unidades de dado no "buffer", do elemento IWU, excede a um dado limiar, a capacidade de banda é renegociada com a rede de forma a incrementar a alocação corrente. O procedimento inverso se aplica quando o "buffer" se esvazia.

9. Fração da capacidade de transmissão do enlace físico, medida em bits por segundo.

Basicamente, este procedimento absorve os surtos de unidades de dado através de um “buffer” no elemento IWU.

Esta implementação apresenta as seguintes desvantagens:

- É necessário um grande “buffer” no IWU para acomodar os surtos de dados provenientes da rede local.
- O tempo de atraso das unidades de dado aumenta em função do tamanho deste “buffer”.
- Sobrecarga de processamento no IWU e nos nós ATM (incluindo os servidores) envolvidos na conexão, em virtude das renegociações de alocação de banda.
- Desperdício de banda nas fases de alocação e desalocação desta.
- A definição dos limiares do “buffer” e do incremento de banda é fortemente dependente do tráfego.

Um procedimento similar a este, para alocação dinâmica de banda, é o protocolo FRP (Fast Reservation Protocol) [29]. Este procedimento, por ser, implementado no nível da camada ATM, reduz a sobrecarga de processamento associado à renegociação de banda.

- C. Alocação com base na banda disponível: Nesta implementação [30] a rede ATM se encarrega de informar periodicamente ao elemento IWU a banda disponível para a conexão semipermanente.

A diferença básica com respeito à implementação apresentada no item B é que a renegociação de banda não é disparada em função do comprimento de um “buffer” e sim do tráfego medido. Dado que o IWU conhece a banda disponível, este pode requerer quantidades coerentes de banda, sem ter que requisitar sucessivos incrementos.

Este procedimento por ser mais eficiente, permite o uso de um “buffer” menor no IWU e conseqüentemente introduzindo um atraso menor para as unidades de dado.

- 2- Conectividade entre os servidores: Esta determina o grau de concentração de tráfego nas conexões semipermanentes e, conseqüentemente, a capacidade de tratamento dos servidores. Assim, no caso de uma baixa conectividade existe uma maior concentração de tráfego nas conexões, implicando nos seguintes efeitos:
- Necessidade de servidores com elevada capacidade de tratamento das unidades de dado.
 - Perfil de tráfego estatisticamente mais previsível e, portanto, uma maior eficiência no procedimento de alocação de banda.
- 3- Número e localização dos servidores: Este aspecto está estreitamente relacionado com o grau de conectividade entre os servidores. Um número reduzido de servidores, apesar de representar uma redução de custos, em geral implica em uma concentração de tráfego nas conexões com os efeitos apresentados acima.

3.3 Elementos de rede

A arquitetura funcional dos elementos de rede envolvidos no suporte do serviço não orientado à conexão é apresentada a seguir:

3.3.1 Elemento Adaptador de Protocolos (IWU)

O elemento adaptador de protocolos IWU pode ser implementado como uma “Bridge”, ou como um “Router”, dependendo do perfil das redes locais que se desejam interconectar.

3.3.1.1 “Bridge”

Esta implementação permite a extensão do serviço de uma rede local a outras remotas, com endereçamento e tamanho de unidades de dado compatíveis. A figura 3.6 apresenta a organização dos protocolos envolvidos nesta implementação.

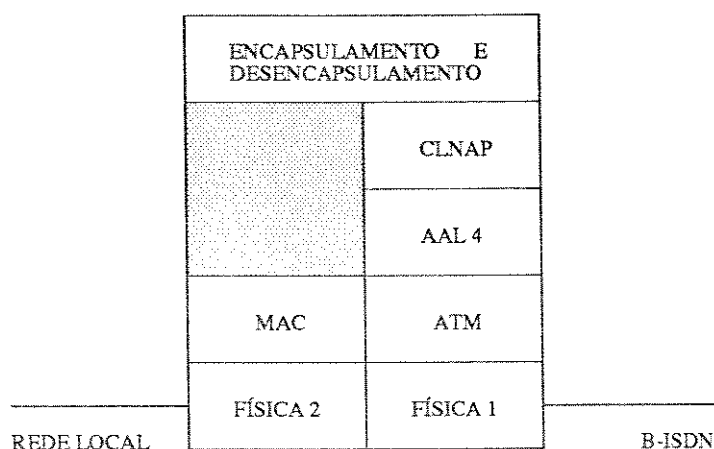


Figura 3.6: IWU implementado como “Bridge”

Na “Bridge” o encaminhamento das unidades de dado entre terminais, locais ou remotos, é realizado através do endereço físico dos terminais, ou seja, do endereço da camada MAC (Medium Access Control) [31], [32], [33] e [34]. Este endereçamento, por exigir que terminais associados a redes locais diferentes tenham endereços físicos diferentes, além de limitar o número máximo de terminais, implica em procedimentos de controle de endereços mais complexos.

De uma forma geral, quando uma unidade de dado é encaminhada de uma Rede Local para a B-ISDN, a “Bridge” encapsula a unidade de dado MAC no campo de informação da unidade de dado CLNAP e traduz os endereços, de origem e de destino, do formato MAC para o formato CLNAP, de forma a endereçar a “Bridge” remota.

No sentido inverso de encaminhamento, a “Bridge” desencapsula a unidade de dado MAC do campo de informação da unidade de dado CLNAP e a transfere ao terminal de destino.

3.3.1.2 “Router”

A implementação do IWU como “Router” se aplica no caso da comunicação entre terminais associados a redes locais não homogêneas, ou seja, que operam com endereçamento e tamanho de unidades de dado não compatíveis. Neste caso, um protocolo de nível de Rede (nível 3 do Modelo OSI) é adicionado à estrutura do IWU para executar as funções de compatibilização entre os formatos de endereço e do tamanho¹⁰ das unidades de dado entre as redes locais. A figura 3.7 apresenta a organização dos protocolos envolvidos nesta implementação.

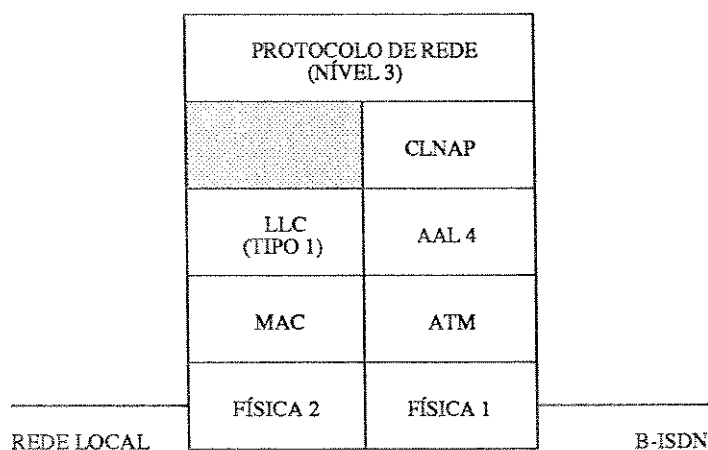


Figura 3.7: IWU implementado como “Router”

Nesta implementação, o endereçamento é realizado no nível do protocolo de rede, não existindo, portanto, as limitações apresentadas pela “Bridge”.

Basicamente, no encaminhamento de uma unidade de dado da rede local para a B-ISDN, o “Router” recupera a unidade de dado do protocolo de rede do campo de informação da unidade de dado LLC¹¹ (Logical Link Control) [35], encapsula esta no campo de informação da unidade de dado CLNAP e traduz os endereços, de origem e de destino (de nível 3), para o formato CLNAP de forma a endereçar o “Router” remoto.

No sentido inverso de encaminhamento o “Router” encapsula a unidade de dado do protocolo de rede no campo de informação da unidade de dado LLC e traduz os endereços, de origem e de destino, de nível 3, para o formato MAC de forma a endereçar o terminal de destino.

Neste processo, o protocolo de rede pode executar as funções de fragmentação e remontagem para compatibilização dos tamanhos das unidades de dado trocados entre as redes locais e destas com a B-ISDN.

10. O tamanho máximo da unidade de dado é função do protocolo MAC e da taxa de transmissão da rede local.

11. Para o serviço não orientado à conexão é especificado o protocolo LLC tipo 1.

3.3.2 Elemento Servidor (CLS)

O elemento servidor CLS (Connectionless Server) possui como funcionalidade básica a execução do protocolo CLNAP. Este pode ser implementado integrando-se as funções dos protocolos CLNAP e AAL 4 em um nó ATM, ou através de um equipamento independente atuando exclusivamente como servidor.

A figura 3.8 representa a estrutura de protocolos do elemento servidor.

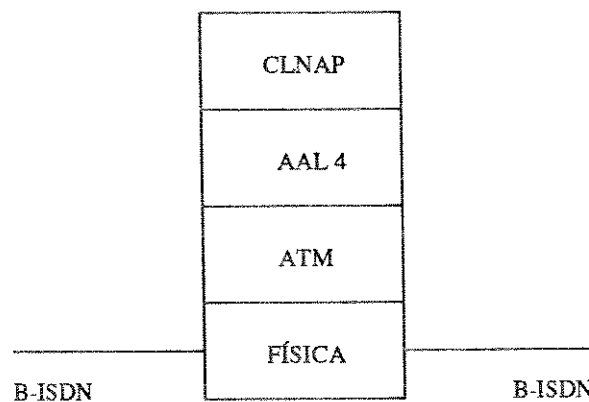


Figura 3.8: Estrutura de protocolos do servidor

O servidor pode executar funções de acesso, ou de trânsito, conforme a sua atuação na rede. O servidor de trânsito é responsável exclusivamente pela função de roteamento das unidades de dado, atuando como um elemento de transporte intermediário entre os servidores de acesso.

Os servidores de acesso são aqueles que se interfaceiam diretamente com os elementos IWU. Estes, além de rotear as unidades de dado, são responsáveis pela função de verificação de compatibilidade dos endereços de origem e destino.

3.3.2.1 Modos de operação do Servidor

As principais funções do elemento servidor são realizadas com base nas informações dos endereços de origem e destino, contidos no cabeçalho da unidade de dado CLNAP. A forma como a camada CLNAP tem acesso a estas informações define o modo de operação do servidor. Os dois modos de operação possíveis são:

- A. Execução das funções da camada CLNAP após a remontagem da unidade de dado CLNAP, ou seja, após a agregação das células ATM que a compõem.
- B. Execução das funções da camada CLNAP após a recepção da primeira célula ATM que compõe a unidade de dado CLNAP. Neste caso, o processamento desta primeira célula deve ser rápido o suficiente de forma a preservar a seqüenciação das demais células da unidade de dado CLNAP.

Este modo de operação baseia-se no fato de que o cabeçalho da unidade de dado CLNAP está inteiramente contido nesta primeira célula conforme mostra a figura 3.9.

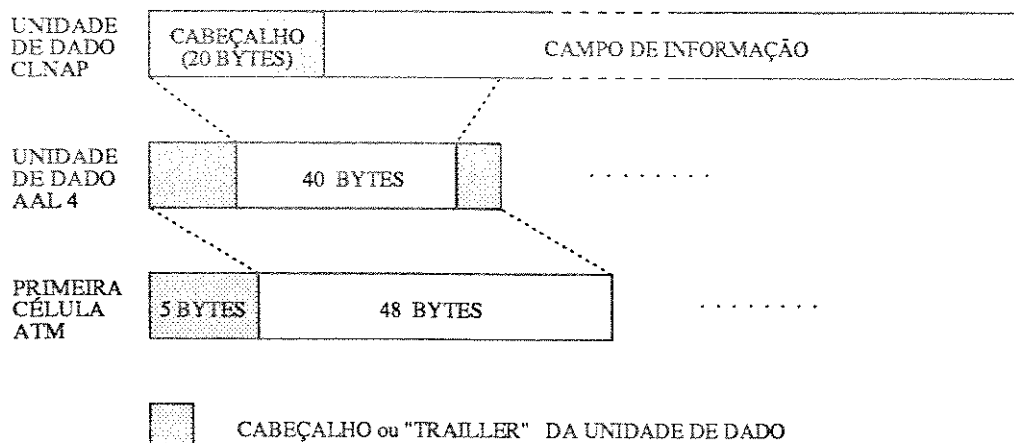


Figura 3.9: Primeira célula da unidade de dado CLNAP

O modo de operação “B”, apesar de ser de implementação mais complexa por operar com restrições de tempo, apresenta as seguintes vantagens em relação ao modo de operação “A”:

- Elevada capacidade de roteamento, por não executar a remontagem das células ATM.
- Reduzida inserção de atraso no processo de roteamento das unidades de dado.
- Desempenho independente do tamanho da unidade de dado CLNAP¹².
- Não armazenamento das células que compõem a unidade de dado CLNAP.

3.4 Conclusão

Este capítulo apresentou as alternativas de arquitetura de rede para o suporte ao serviço não orientado à conexão, identificando os elementos de rede IWU e CLS envolvidos e uma análise das opções de implementação destes elementos.

O cenário Direto e as implementações dos elementos IWU, como “Router”, e CLS sem executar a remontagem das células ATM, são apontados como as opções mais genéricas e adotadas como hipóteses no desenvolvimento dos próximos capítulos.

12. Este tamanho pode variar desde 24 bytes até 9232 bytes.

Capítulo 4

Funcionalidades dos elementos de rede

A seguir é apresentado um detalhamento das funcionalidades dos elementos IWU e CLS, concentrando-se nos protocolos dedicados ao serviço não orientado à conexão, ou seja, AAL 4, CLNAP e IP (Internet Protocol).

Este detalhamento está baseado nas seguintes hipóteses, que representam uma abordagem abrangente para a descrição das funcionalidades dos elementos de rede.

1. Adoção do cenário direto.
2. Elemento IWU implementado como “Router” e utilizando o protocolo de rede IP.
3. Elemento CLS operando após a recepção da primeira célula da unidade de dado CLNAP.

4.1 Camada de Adaptação AAL 4

A camada AAL 4 [21] está organizada nas subcamadas SAR (Segmentation and Reassembly) e CS (Convergence Sublayer), sendo esta última subdividida, ainda, nas subcamadas SSCS (Service Specific Convergence Sublayer) e CPCS (Common Part Convergence Sublayer)¹³.

A figura 4.1 ilustra a organização da camada AAL 4 em subcamadas.

13. As subcamadas SAR e CPCS são as mesmas para a AAL 3 e AAL 4.

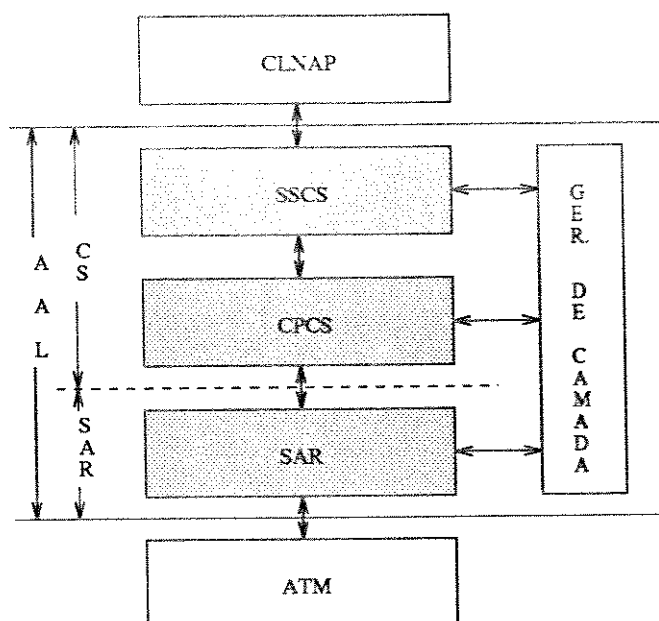


Figura 4.1: Organização da camada AAL 4

4.1.1 Funções

As funções da camada AAL 4 estão implementadas de forma distribuída nas subcamadas SAR, CPCS e SSCS.

4.1.2 Funções da subcamada SAR

As funções da subcamada SAR, implementadas nos elementos IWU e CLS, são as seguintes:

- A- Segmentação e remontagem: Estas se aplicam, respectivamente, nos processos de transmissão e recepção quando a unidade CPCS_PDU (CPCS_Protocol Data Unit)¹⁴ é maior do que o campo de informação da unidade de dado SAR (SAR_PDU)¹⁵.
- B- Detecção de erro de bit: A integridade da unidade SAR_PDU é verificada através do cálculo de um CRC (Cyclic Redundancy Check).
- C- Detecção de perda ou de inserção de SAR_PDU: Esta corresponde a verificação da integridade da sequência das unidades SAR_PDU.

14. A unidade CPCS_PDU pode ser formada por uma ou mais unidades SAR_PDU.

15. A unidade SAR_PDU é transportada no campo de informação da célula ATM (ATM_PDU).

- D- Inserção e extração de preenchimento: Estas se aplicam, respectivamente, nos processos de transmissão e recepção, quando o tamanho da unidade CPCS_PDU não é múltiplo do tamanho do campo de informação da SAR_PDU (44 bytes).

Dentre estas funções, a remontagem e a segmentação não são implementadas no elemento CLS, devido a terceira hipótese apresentada no início deste capítulo.

4.1.2.1 Estrutura da SAR PDU

Para a implementação das funções, acima mencionadas, a unidade SAR_PDU foi estruturada conforme a figura 4.2.

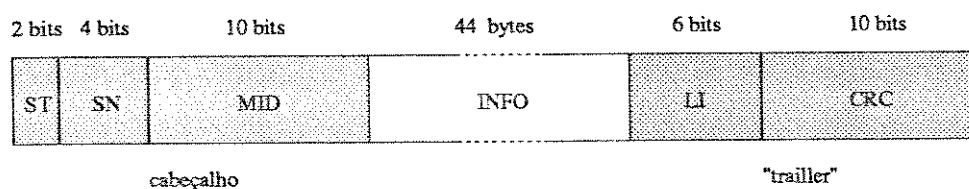


Figura 4.2: Estrutura da SAR_PDU

onde,

- ST (Segment Type): Identifica a unidade SAR_PDU conforme a tabela 4.1.

TIPO DE SAR_PDU	CÓDIGO	SIGNIFICADO
BOM	10	PRIMEIRA SAR_PDU DA CPCS_PDU
COM	00	SAR_PDU INTERMEDIÁRIA
EOM	01	ÚLTIMA SAR_PDU DA CPCS_PDU
SSM	11	CPCS_PDU INTEIRAMENTE CONTIDA NA SAR_PDU

Tabela 4.1: Identificação da SAR_PDU

- SN (Sequence Number): Especifica a posição relativa (módulo 16) das unidades SAR_PDU que compõem uma unidade CPCS_PDU.
- MID¹⁶ (Multiplexing Identification): Identifica a unidade CPCS_PDU, de forma que todas as unidades SAR_PDU pertencentes a esta unidade possuem o campo MID com o mesmo valor. Este valor deve ser diferenciado para unidades CPCS_PDU transmitidas consecutivamente para que a recepção efetue o processo de remontagem adequadamente.
- LI (Length Indication): Indica, no formato binário, o número de bytes transportados no campo de informações da unidade SAR_PDU. A tabela 4.2 indica os valores possíveis para o campo LI de acordo com o tipo de SAR_PDU.

16. Esta sigla se deve ao fato de que para a AAL 3 este campo é usado para definir mais um nível de multiplexação na conexão ATM.

TIPO DE SAR_PDU	VALOR DO CAMPO LI
BOM	44 BYTES
COM	44 BYTES
EOM	4 a 44 BYTES
SSM	8 a 44 BYTES

Tabela 4.2: Valores possíveis para o campo LI

- CRC (Cyclic Redundancy Check): Utilizado para se detetar erros no transporte da unidade SAR_PDU. Este se aplica aos demais campos da unidade SAR_PDU.
- INFO (Information): Este campo é utilizado para o transporte da unidade CPCS_PDU.

A figura 4.3 ilustra a relação entre as unidades SAR_PDU e CPCS_PDU nos processos de segmentação (visto de cima para baixo) e remontagem (visto de baixo para cima).

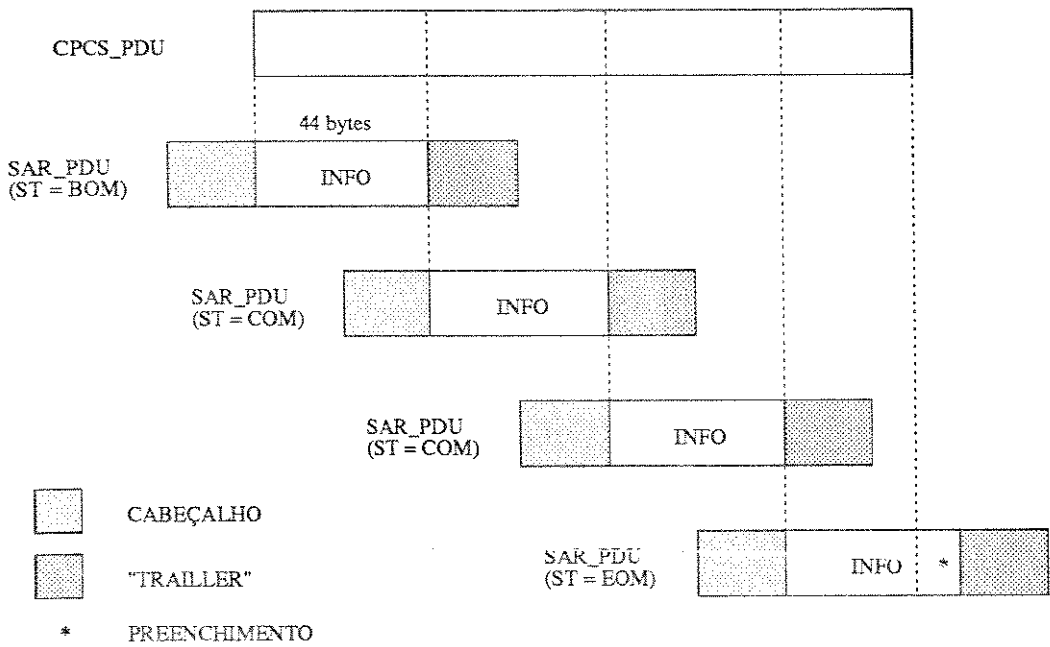


Figura 4.3: Processos de segmentação e remontagem

4.1.3 Funções da subcamada CPCS

A subcamada CPCS executa as seguintes funções:

- A- Verificação da integridade da CPCS_PDU: Esta função corresponde a uma validação do processo de remontagem, verificando se todas as unidades SAR_PDU de uma CPCS_PDU foram recebidas corretamente.
- B- Alocação de memória: Esta função consiste em indicar de forma antecipada à recepção, através da primeira unidade SAR_PDU, o tamanho da CPCS_PDU a que ela pertence antes do término do processo de remontagem.
- C- Inserção e extração de preenchimento: Estas se aplicam, respectivamente, nos processos de transmissão e recepção quando o tamanho do campo de informação da unidade CPCS_PDU não é múltiplo de quatro bytes. Esta multiplicidade é exigida para otimizar a implementação.
Para o serviço não orientado à conexão esta função não é executada, visto que a camada CLNAP garante esta multiplicidade.

4.1.3.1 Estrutura da CPCS_PDU

A unidade CPCS_PDU está estruturada conforme a figura 4.4.

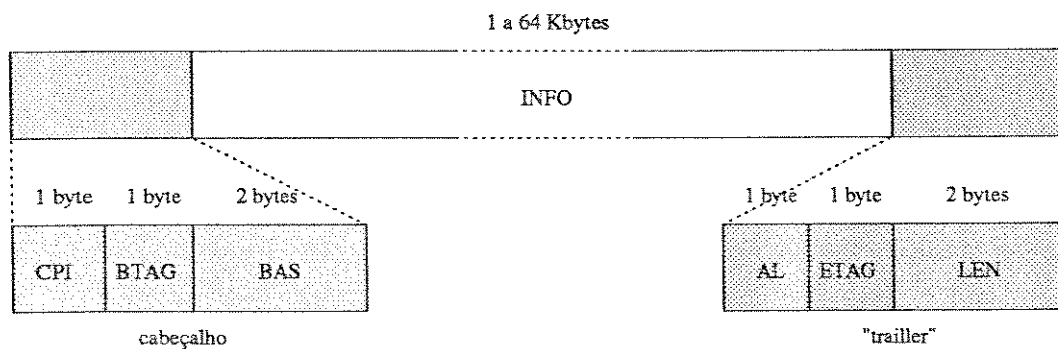


Figura 4.4: Estrutura da CPCS_PDU

onde,

- BTAG e ETAG (Beginning Tag e End Tag): Estes dois campos são gerados com o mesmo valor para uma dada CPCS_PDU, devendo ser diferenciados para unidades CPCS_PDU transmitidas consecutivamente.
Estes campos são comparados na recepção, para se prevenir da situação de perda da última e da primeira SAR_PDU de unidades CPCS_PDU consecutivas e de mesmo tamanho.
- LEN (Lenght): Apresenta o tamanho do campo de informação da CPCS_PDU. No processo de remontagem o conteúdo deste campo deve corresponder a somatória do conteúdo do campo LI das unidades SAR_PDU.
- BAS (Buffer Allocation Size): Este campo apresenta o tamanho do campo de informação da CPCS_PDU, que pode ser usado para antecipar o processo de alocação de memória na recepção.
- CPI (Common Part Indicator): Indica a unidade de referência para os campos BAS e LEN. Até agora somente o valor zero foi previsto, implicando que a unidade de referência dos campos BAS e LEN é o byte.

- AL (Alignment): Campo de conteúdo zero usado para a função de preenchimento.
- INFO (Information): Campo de informação para transporte da unidade SSCS_PDU.

4.1.4 Funções da subcamada SSCS

A subcamada SSCS é responsável pela adaptação do serviço do usuário a subcamada CPCS. Assim, de acordo com o tipo de serviço requerido à subcamada SSCS pode implementar, por exemplo, funções de segmentação e remontagem, “blocking” e “deblocking”¹⁷, controle de fluxo e recuperação de erros.

No suporte ao serviço não orientado à conexão, a subcamada SSCS não é implementada visto que a camada IP executa as funções de segmentação e remontagem e as demais funções complementares são executadas pelo equipamento do usuário.

Assim, as unidades CLNAP_PDU correspondem as unidades CPCS_SDU (CPCS Service Data Unit), ou seja, são mapeadas diretamente no campo de informação das unidades CPCS_PDU.

4.1.5 Primitivas associadas à camada AAL 4

A seguir, são apresentadas as primitivas e seus parâmetros, considerados neste trabalho, que interagem com as camadas CLNAP e ATM.

4.1.5.1 Primitivas de interação com a camada CLNAP

- A. AAL_DATA_request (AAL_SDU)¹⁸: Esta é usada pela camada CLNAP para requerer a transferência de uma unidade CLNAP_PDU através de uma conexão AAL.
- B. AAL_DATA_indication (AAL_SDU): Esta primitiva é usada para transferência de uma unidade AAL_SDU, recebida sem erros, para a camada CLNAP através de uma conexão AAL.

O parâmetro AAL_SDU destas primitivas corresponde a unidade CLNAP_PDU.

4.1.5.2 Primitivas de interação com a camada ATM

- A. ATM_DATA_request (ATM_SDU)¹⁹: Esta é usada pela camada AAL para requerer a transferência de uma unidade SAR_PDU através de uma conexão ATM.

17. As funções de “blocking” e “deblocking” aplicam-se, respectivamente, nos processos de transmissão e recepção, quando o tamanho das unidades de dados do usuário é pequeno e necessita ser agregado em unidades maiores antes da transmissão.

18. Os parênteses, aqui, são utilizados para a definição dos parâmetros da primitiva.

19. O parâmetro CLP (Cell Loss Priority), que define o nível de prioridade da célula ATM não é considerado neste trabalho.

- B. ATM_DATA_indication (ATM_SDU)²⁰: Esta primitiva é usada para transferência de uma unidade ATM_SDU, recebida sem erros, para a camada AAL através de uma conexão ATM.

O parâmetro ATM_SDU destas primitivas corresponde a unidade SAR_PDU.

20. O parâmetro CI (Congestion Indication), que indica a ocorrência de congestionamento na rede, não é considerado neste trabalho.

4.2 Camada CLNAP

A camada CLNAP [25] é a responsável pelas principais funções envolvidas no suporte ao serviço não orientado à conexão na B-ISDN. Esta camada, conforme representado na figura 4.5, interage com a camada AAL 4, com a gerência de camada e com a camada IP (no caso do elemento IWU).

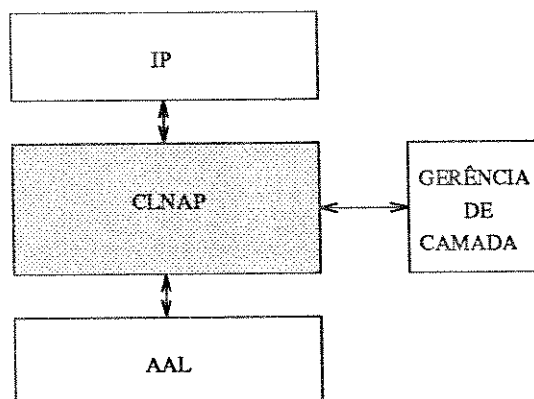


Figura 4.5: Interfaces com a camada CLNAP

4.2.1 Funções da camada CLNAP

As funções implementadas pela camada CLNAP estão distribuídas nos elementos IWU e CLS, de acordo com a atuação destes na rede. Estas funções são descritas a seguir:

- A- Roteamento: Esta função é intrínseca ao elemento CLS, podendo também se aplicar ao elemento IWU, quando existe mais de uma opção de conexão semipermanente, entre este e a rede virtual, para o encaminhamento da unidade CLNAP_PDU.

A seleção da conexão, aqui designada por AAL_CI (AAL Connection Identification), é feita baseada no endereço de destino e na classe de qualidade de serviço²¹, especificados para o transporte da unidade CLNAP_PDU.

As associações dos endereços de destino e das classes de qualidade de serviço com as conexões AAL são efetuadas na ocasião do estabelecimento das conexões semipermanentes. Estas associações podem, por exemplo, ser representadas pelas tabelas 4.3 e 4.4.

21. As classes de qualidade de serviço podem ser definidas, por exemplo, em função do tempo de atraso, prioridade, confiabilidade e tráfego desejados para o transporte da unidade IP_PDU.

ENDEREÇO DE DESTINO (DA)	CONEXÃO AAL
DA 1	AAL_CI i
	AAL_CI j
	etc
DA 2	AAL_CI m
	AAL_CI n
	etc
etc	etc

Tabela 4.3: Opções de conexões AAL por endereço de destino

CLASSE DE QUALIDADE DE SERVIÇO (QoS)	CONEXÃO AAL
QoS 1	AAL_CI p
	AAL_CI q
	etc
QoS 2	AAL_CI r
	AAL_CI s
	etc
etc	etc

Tabela 4.4: Opções de conexões AAL por classe de qualidade de serviço

Assim uma unidade CLNAP_PDU teria como opção de encaminhamento o resultado da intersecção dos conjuntos de AAL_CI associados ao endereço de destino e a classe de qualidade de serviço especificados para esta unidade. No caso de o resultado desta intersecção ser nulo, a unidade CLNAP_PDU pode ser descartada e se existir mais de uma opção de encaminhamento qualquer uma delas pode ser adotada.

Estas tabelas podem ser atualizadas automaticamente, por exemplo, em ocasiões de queda do enlace físico, queda da conexão semipermanente ou em situações de congestionamento da rede.

- B- Replicação: Esta função, implementada no elemento CLS, é decorrente do endereçamento de grupo requisitado pelo usuário para o transporte da unidade IP_PDU. Esta função implica, ao se constatar que o endereço de destino é um endereço de grupo²², em replicar quando necessário, a unidade CLNAP_PDU.

A verificação da necessidade e do número de replicações é feita com base na análise dos endereços individuais²³ representados pelo endereço de grupo e na topologia da rede virtual não orientada à conexão. A tabela 4.5 representa a decomposição do endereço de grupo em endereços individuais.

22. Endereço de grupo, ou "multicast", é aquele que representa mais de um endereço de destino.

23. Endereço individual é aquele que representa um único endereço.

ENDEREÇO DE GRUPO	ENDEREÇO INDIVIDUAL
DA_g i	DA m
	DA n
	etc
DA_g j	DA p
	DA q
	etc
etc	etc

Tabela 4.5: Decomposição do endereço de grupo

- C- Validação de endereços: Esta função, implementada no elemento CLS de acesso, consiste em verificar os dados de cadastro referentes as unidades CLNAP_PDU recebidas dos elementos IWU filiados ao CLS. Esta verificação é feita pela análise do endereço de origem da unidade CLNAP_PDU .
- D- Filtro de endereços: Esta função, executada no elemento CLS de acesso, consiste em verificar se a unidade CLNAP_PDU recebida pode ser encaminhada para o endereço de destino especificado, ou seja, ao elemento IWU de destino. Esta função baseia-se na análise de compatibilidade dos endereços de origem e destino conforme representado na tabela 4.6.

ENDEREÇO DE ORIGEM (SA)	ENDEREÇO DE DESTINO (DA)
SA i	DA j
	DA k
	etc
SA j	DA m
	DA n
	etc
etc	etc

Tabela 4.6: Restrição de encaminhamento

Esta tabela é diferenciada para cada CLS de acesso de forma a se constituir domínios de restrições de acesso. Um particular de domínio é o grupo fechado de usuários, onde cada usuário, neste trabalho, é uma rede local.

- E- Detecção de erro: Esta função, implementada somente no elemento IWU²⁴, é realizada através do cálculo de um CRC. Neste trabalho o uso desta função está condicionado à classe de qualidade de serviço especificada para cada unidade CLNAP_PDU.
- F. Inserção e extração de preenchimento: Estas funções são executadas no elemento IWU²⁴, respectivamente, nos processos de transmissão e recepção, quando o tamanho do campo de informação da unidade CLNAP_PDU não é múltiplo de quatro bytes.

A definição e atualização das tabelas 4.3, 4.4, 4.5 e 4.6 são realizadas por procedimentos de gerência de rede através de primitivas da interface com a gerência de camada.

4.2.1.1 Estrutura da CLNAP_PDU

A unidade CLNAP_PDU está estruturada conforme a figura 4.6.

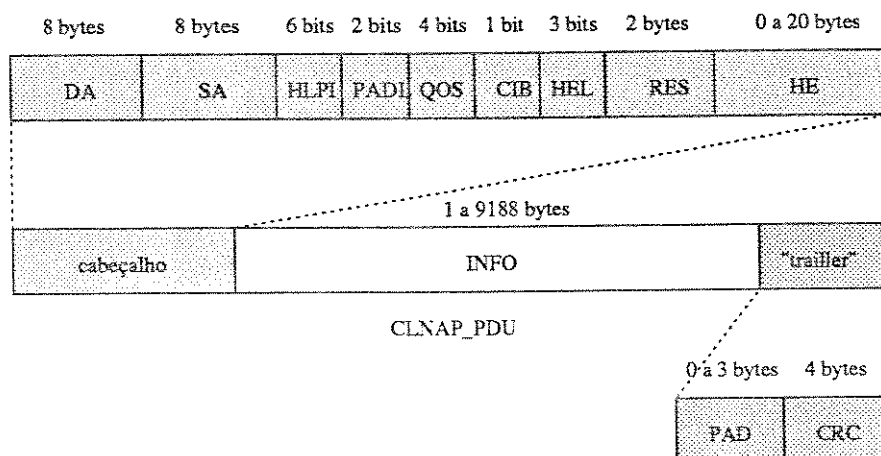


Figura 4.6: Estrutura da CLNAP_PDU

onde,

- DA e SA (Destination Address e Source Address): Representam, respectivamente, os endereços de destino e origem. Nestes campos os primeiros quatro bits especificam se o endereço é individual ou de grupo e os demais bits correspondem à codificação, no formato E.164 [36], do endereço propriamente dito. A tabela 4.7 apresenta a estrutura para os campos DA e SA.

24. Esta função não é implementada no elemento CLS, visto que este não executa a função de remontagem das unidades SAR_PDU.

ENDEREÇO DE DESTINO		ENDEREÇO DE ORIGEM	
TIPO	ENDEREÇO	TIPO	ENDEREÇO
1100	INDIVIDUAL	1100	INDIVIDUAL
1110	GRUPO	*	—

* não se aplica

Tabela 4.7: Estrutura dos campos DA e SA

- HLPI (Higher Layer Protocol Identifier)²⁵: Especifica para qual protocolo de nível superior a unidade CLNAP_PDU recebida será encaminhada. Neste trabalho este campo deve conter o código do protocolo IP.
- PADL (Padding Length): Este campo indica o número de bytes, de 0 a 3, adicionados ao campo de informação para que o tamanho deste se torne múltiplo de quatro bytes.
- QOS (Quality of Service)²⁶: Designa a classe de qualidade de serviço desejada para o transporte da unidade CLNAP_PDU.
- CIB (CRC Indication Bit): Indica a presença (CIB = 1) ou ausência (CIB = 0) do campo CRC na unidade CLNAP_PDU. Este bit pode ser definido em função da classe de qualidade de serviço desejada para o transporte da unidade CLNAP_PDU.
- HEL (Header Extension Length): Este campo indica o tamanho da extensão do cabeçalho, em unidades de quatro bytes. Para este trabalho o conteúdo deste campo é zero.
- RES (Reserved): A função deste campo ainda não está definida, assumindo provisoriamente o valor zero.
- HE (Header Extension): Campo reservado para a expansão do cabeçalho.
- INFO (Information): Campo, de tamanho variável, usado para o transporte da unidade IP_PDU.
- PAD (Padding): São bytes (0 a 3) de conteúdo zero utilizados para executar a função de preenchimento.
- CRC (Cyclic Redundancy Check): Campo opcional utilizado para se detetar erros no transporte da unidade CLNAP_PDU. Este se aplica aos demais campos da CLNAP_PDU considerando que o campo RES tem conteúdo zero. Detalhes da geração e verificação deste campo podem ser encontrados em [37].

4.2.2 Primitivas associadas à camada CLNAP

Este item descreve as primitivas trocadas com a camada IP para o elemento IWU.

25. Esta codificação ainda não está definida.

26. O conteúdo deste campo ainda não está definido.

4.2.2.1 Primitivas de interação com a camada IP

- A. CLNAP_DATA_request (SA,DA,QOS,CLNAP_SDU): Esta é usada pela camada IP para requerer a transferência de uma unidade IP_PDU para a camada CLNAP.
- B. CLNAP_DATA_indication (SA,DA,QOS,CLNAP_SDU): Esta primitiva é usada para transferência de uma unidade CLNAP_SDU, recebida sem erros, para a camada IP.

O parâmetro CLNAP_SDU destas primitivas corresponde a unidade IP_PDU e os demais parâmetros estão descritos no item 4.2.1.1.

4.3 Camada IP

A camada IP, [17] e [38], é a estrutura de mais alto nível implementada no elemento IWU e é responsável pela compatibilização entre as redes locais de origem e de destino, e destas com a B-ISDN. A figura 4.7 apresenta as interfaces associadas à camada IP.

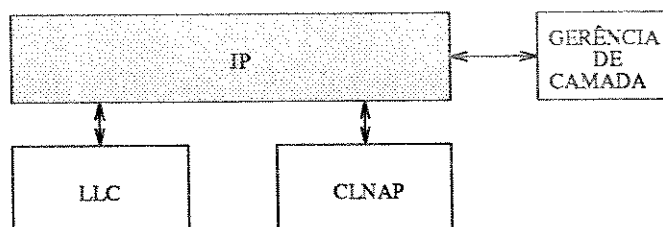


Figura 4.7: Interfaces com a camada CLNAP

4.3.1 Funções da camada IP

A camada IP é responsável pelas seguintes funções:

- A- Remontagem e fragmentação: Estas se aplicam tanto no sentido de transmissão (rede local para B-ISDN) quanto na recepção (B-ISDN para rede local), compatibilizando respectivamente o tamanho da LLC_PDU com a CLNAP_PDU e vice-versa.

O tamanho máximo da IP_PDU suportado pela B-ISDN é de 9.188 bytes, que corresponde ao limite do campo de informação da unidade CLNAP_PDU. Para a rede local este tamanho máximo é dependente de parâmetros específicos desta, tais como, o método de acesso ao meio compartilhado²⁷ (camada MAC) utilizado, a velocidade de operação da rede e as características de desempenho desejadas. Desta forma, considera-se que o tamanho máximo da unidade IP_PDU suportado pela rede local é limitado pela camada IP, ou seja, 64 Kbytes.

A função de remontagem é sempre praticada devido ao fato do protocolo IP não permitir a fragmentação de fragmentos e, também, em virtude do processo de fragmentação poder gerar, dependendo da implementação realizada no terminal de origem, fragmentos de tamanhos variáveis.

No caso do processo de fragmentação gerar fragmentos de tamanhos uniformes (exceto o último), a função de remontagem poderia não ser necessária se o tamanho do primeiro fragmento fosse compatível com a rede destino. Esta situação, porém, não é considerada em virtude do fato do serviço não orientado à conexão não garantir a seqüenciação dos fragmentos que compõem a unidade IP_PDU.

A figura 4.8 ilustra os processos de remontagem e fragmentação realizados, pelos IWU de origem e destino, na interconexão de duas redes locais que operam com unidades de dado de tamanhos diferentes.

27. Por exemplo: "Token ring", "Token bus" e CSMA/CD.

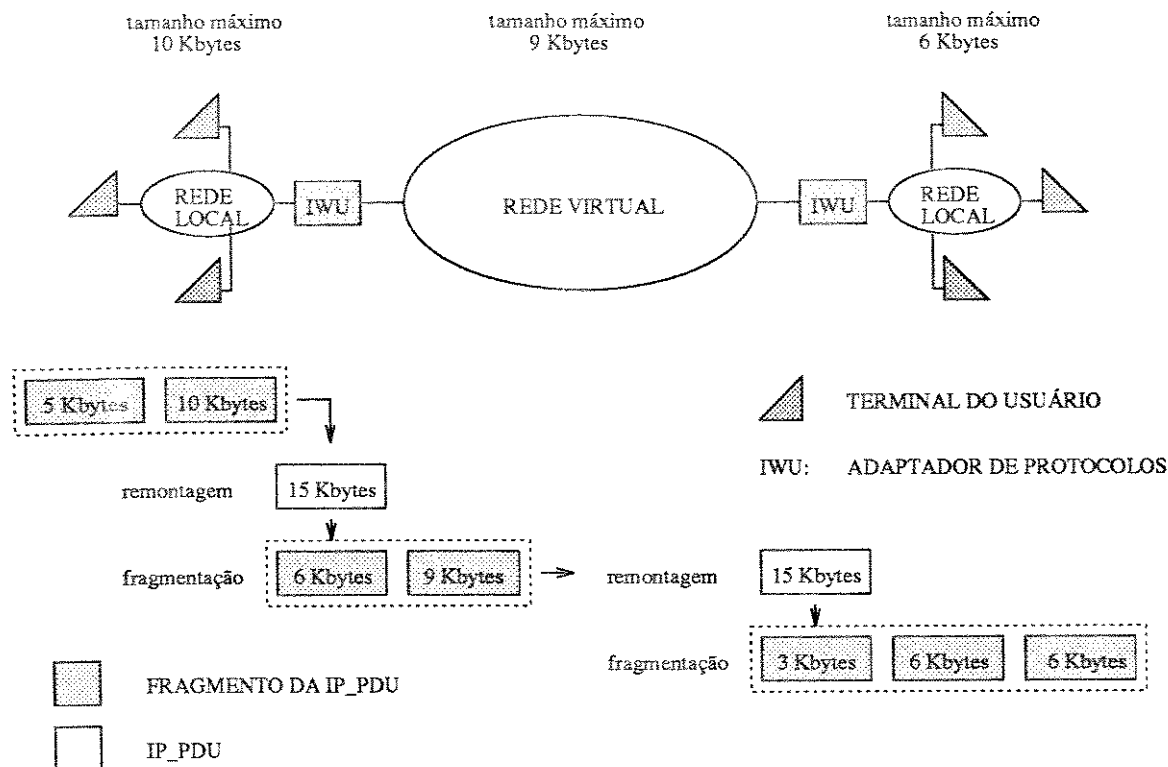


Figura 4.8: Processos de remontagem e fragmentação de unidades IP_PDU, considerando unidades de tamanhos arbitrários

- B- **Conversão de endereços:** Esta função é responsável pela conversão do endereço no formato IP para o formato E.164 no sentido de transmissão (rede local para B-ISDN), e para o formato MAC no sentido de recepção (B-ISDN para rede local).

Nesta conversão está implícita a manipulação de endereços individuais para designação de origem e destino, e de endereços de grupo para designação de destino. O endereço de grupo é aquele que representa um conjunto de endereços individuais, de modo que uma unidade IP_PDU com este tipo de endereço deve ser encaminhada para um conjunto de destinos especificados por seus endereços individuais.

As tabelas 4.8, 4.9 e 4.10 representam os dados necessários para a execução do mapeamento de endereços pelo elemento IWU. A definição e atualização destas tabelas são realizadas por procedimentos de gerência de rede através de primitivas da interface com a gerência de camada.

ENDEREÇO IP	ENDEREÇO MAC
IP 1	MAC a
IP 2	MAC b
IP 3	MAC c
etc	etc

Tabela 4.8: Associação entre os endereços IP e MAC de uma rede local

ENDEREÇO IP	ENDEREÇO E.164
IP 1	E.164 i
IP 2	
IP 3	
etc	
IP 4	E.164 n
IP 5	
IP 6	
etc	
etc	etc

Tabela 4.9: Associação dos endereços IP, por rede local, com os endereços de rede E.164 correspondentes

ENDEREÇO DE GRUPO IP	ENDEREÇO DE GRUPO E.164	ENDEREÇO DE GRUPO MAC
IP_g i	E.164_g i	MAC_g l
IP_g j	E.164_g j	MAC_g m
IP_g k	E.164_g k	MAC_g n
etc	etc	etc

Tabela 4.10: Associação entre os endereços de grupo IP, MAC e E.164

As tabelas 4.8 e 4.9 são utilizadas, respectivamente, nos sentidos de recepção e transmissão no mapeamento de endereços individuais. A tabela 4.8 é diferenciada para cada elemento IWU e representa a associação entre os endereços IP e MAC dos terminais que formam a rede local. A tabela 4.9, comum aos elementos IWU das redes locais que desejam ser interconectadas, relaciona os conjuntos de endereços IP dos terminais das redes locais com os respectivos endereços de rede E.164. A figura 4.9 ilustra a manipulação das tabelas 4.8 e 4.9 no mapeamento de endereços individuais.

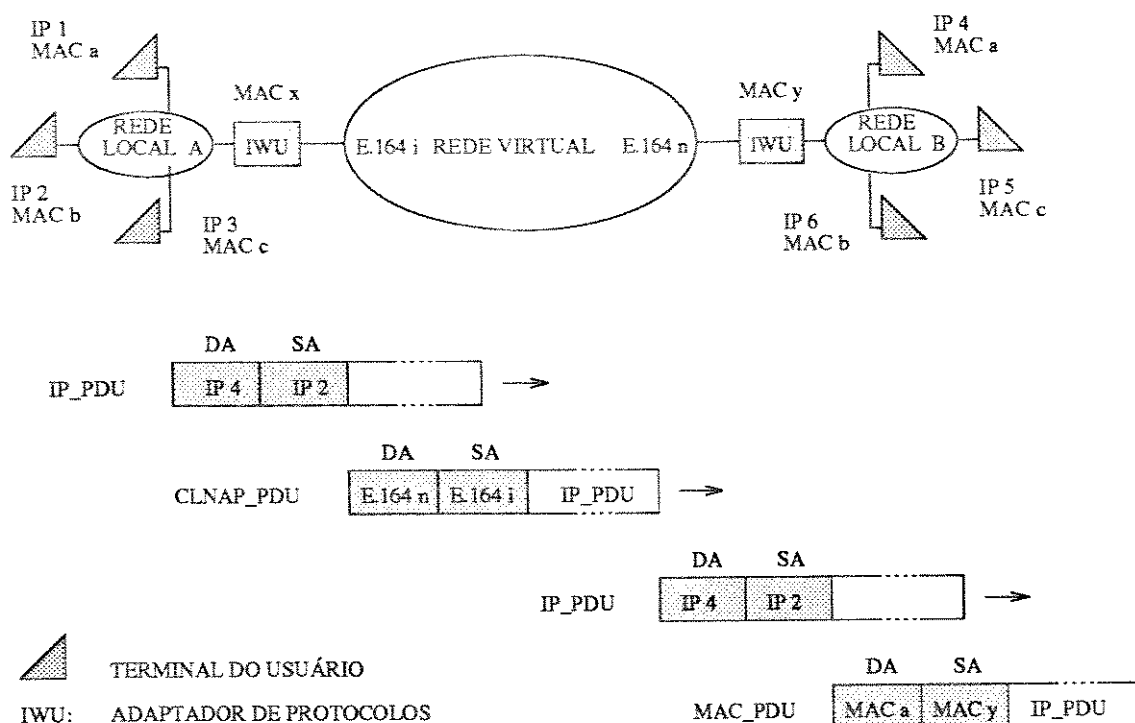
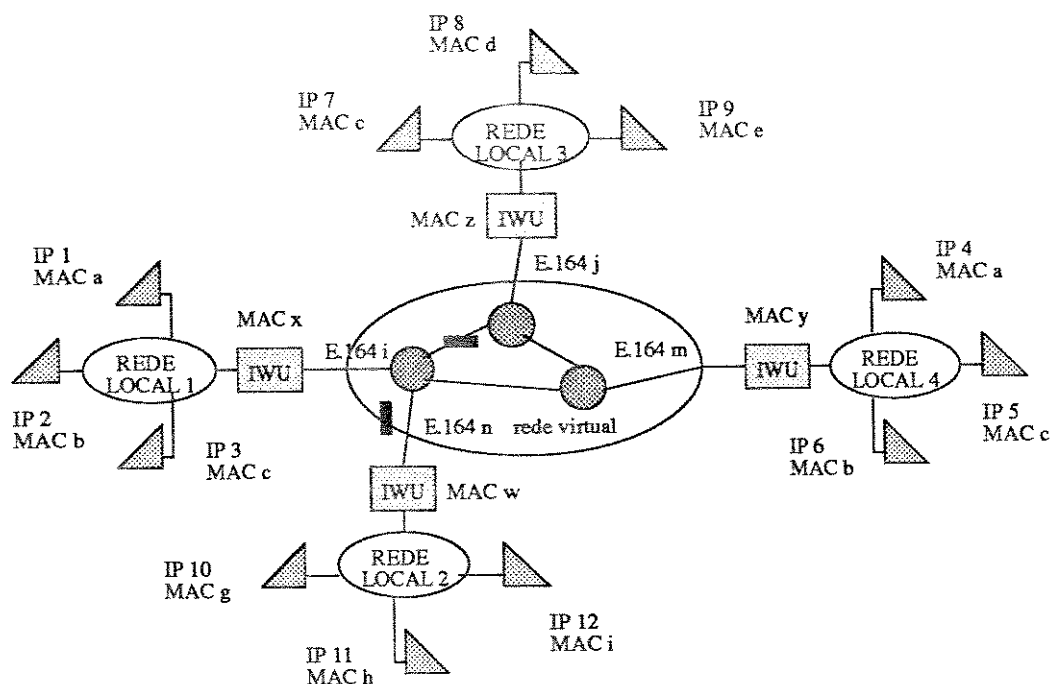


Figura 4.9: Mapeamento de endereços individuais. Note que para a rede local “A” existe uma correspondência com as tabelas 4.8 e 4.9

A tabela 4.10 é utilizada no mapeamento de endereços de grupo, associando para cada endereço de grupo no formato IP os endereços de grupo nos formatos E.164 e MAC. Os endereços MAC desta tabela são diferenciados para cada elemento IWU, em função dos terminais a serem acessados na rede local.

A figura 4.10 ilustra a formação de um endereço do grupo, “IP_g i”, que representa seis terminais alocados às redes locais 1, 2 e 3. Esta figura também exemplifica o processo de replicação, no elemento CLS, de uma unidade IP_PDU transmitida pela rede local 1.



$IP_g\ i = \{IP\ 1, IP\ 2, IP\ 10, IP\ 11, IP\ 7, IP\ 8\}$

$E.164_g\ i = \{E.164\ i, E.164\ j, E.164\ n\}$

$MAC_g\ 1 = \{MAC\ a, MAC\ b\}$

$MAC_g\ 2 = \{MAC\ g, MAC\ h\}$

$MAC_g\ 3 = \{MAC\ c, MAC\ d\}$

IWU: ADAPTADOR DE PROTOCOLOS

TERMINAL DO USUÁRIO

SERVIDOR

IP_PDU
(com endereço de grupo $IP_g\ i$)

Figura 4.10: Formação de um endereço de grupo

- C- Definição do serviço de transporte: Esta função consiste na análise dos atributos de transporte, especificados para a unidade IP_PDU, através do campo TOS (Type of Service) e tradução no parâmetro de classe de serviço²⁸ da camada LLC [35] ou no parâmetro de classe de qualidade de serviço da camada CLNAP (campo QOS). Estes parâmetros qualificam, respectivamente, o serviço de transporte na rede local e na B-ISDN.
- D- Validação da unidade IP_PDU: Esta função é realizada nos sentidos de transmissão e recepção, e se baseia nas seguintes verificações:
- Integridade do cabeçalho da unidade IP_PDU, através da análise do campo CRC.
 - Versão do protocolo IP utilizada, através da análise do campo VER (Version).
 - Tempo de vida da unidade IP_PDU, através da análise do campo TTL (Time to Live).

28. O parâmetro classe de serviço define o nível de prioridade desejado para o transporte da unidade LLC_PDU na rede local.

4.3.1.1 Estrutura da IP_PDU

A figura 4.11 representa a estrutura da unidade IP_PDU.

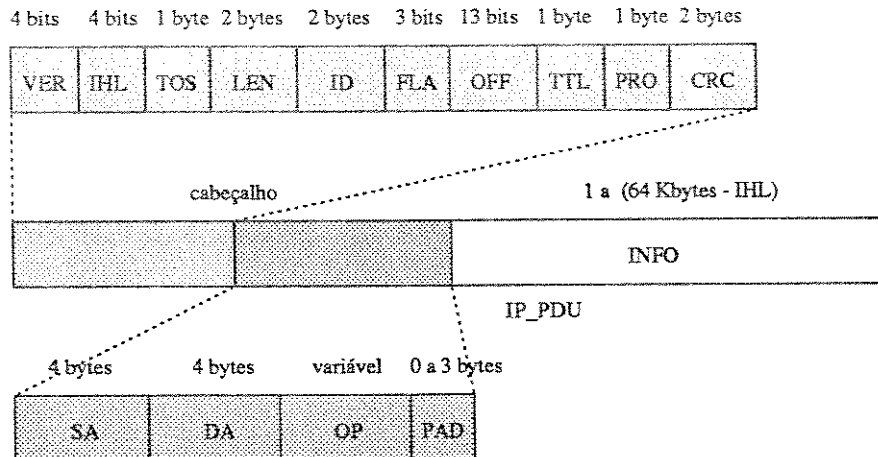


Figura 4.11: Estrutura da IP_PDU

onde,

- VER (Version): Este campo, incluído para permitir a evolução do protocolo, especifica o número da versão do protocolo IP utilizado.
- IHL (Internet Header Length): Indica, em unidades de quatro bytes, o tamanho do cabeçalho da unidade IP_PDU. O valor mínimo para este campo é cinco, ou seja, o tamanho mínimo do cabeçalho é de vinte bytes.
- TOS (Type of Service): Designa os níveis de prioridade, confiabilidade, atraso e tráfego desejados para o transporte da unidade IP_PDU.
- LEN (Length): Indica o tamanho, em bytes, da unidade IP_PDU.
- ID (Identification): Este, conjuntamente com os campos DA, SA e PRO (Protocol), é utilizado para identificar univocamente uma unidade IP_PDU ou os fragmentos que a compõem. O terminal de origem deve garantir²⁹ esta unicidade de identificação durante o tempo necessário para o encaminhamento, da unidade IP_PDU, até o seu destino.
- FLA (Flags): Este campo especifica se a unidade IP_PDU pode ser fragmentada (bit ENABLE = 1) e, no caso desta ser um fragmento, se é o último (bit MORE = 0).
- OFF (Offset): Indica a posição relativa, em unidades de oito bytes, do fragmento dentro da unidade IP_PDU original. Isto implica que o tamanho do campo de informação (INFO) dos fragmentos, com exceção do último, deve ser múltiplo de oito bytes.

A tabela 4.11 ilustra a manipulação dos campos FO e FLA no processo de fragmentação de uma unidade IP_PDU.

29. Os 16 bits do campo ID permite o número máximo de 65.536 unidades IP_PDU, com mesma origem e destino, em trânsito na rede.

IP_PDU	COMPRIMENTO DO CAMPO "INFO"	bit "MORE"	CAMPO "OFF"
ORIGINAL	472 BYTES	0	0
PRIMEIRO FRAGMENTO	240 BYTES (1)	1 (3)	0
SEGUNDO FRAGMENTO	232 BYTES (2)	0	30 (4)

(1) TAMANHO MÁXIMO PARA O FRAGMENTO

(2) 472 - 240

(3) INDICAÇÃO DE QUE AINDA EXISTE FRAGMENTO

(4) 240/8

Tabela 4.11: Manipulação dos campos OFF e FLA no processo de fragmentação

- TTL (Time to Live): Indica o tempo, em segundos, que resta para o transporte da unidade IP_PDU. Os roteadores podem, por exemplo, decrementar o conteúdo deste campo e descartar a unidade IP_PDU caso este tenha valor zero. Este procedimento impede que uma unidade IP_PDU, por algum motivo, fique em "loop" na rede.
- PRO (Protocol): Especifica para qual protocolo de nível superior a unidade IP_PDU recebida será encaminhada. Este campo, para este trabalho, só é analisado pelo terminal do usuário.
- CRC (Cyclic Redundancy Check): Este campo é utilizado para se detectar erros no cabeçalho³⁰ ocorridos durante o transporte da unidade IP_PDU. A detecção de erro implica no descarte da IP_PDU.
- DA e SA (Destination Address e Source Address): Estes campos representam, respectivamente, os endereços de destino e origem. Nestes campos os primeiros quatro bits especificam, de forma equivalente à apresentada para a camada CLNAP, se o endereço é individual ou de grupo. Os demais bits correspondem à codificação, no formato IP, do endereço propriamente dito.
- OP (Options): Campo opcional de tamanho variável, definido pelo terminal de origem, que especifica as opções para o transporte da unidade IP_PDU, tais como, o nível de segurança (ex. comum, confidencial, secreto) e o algoritmo de roteamento.
- PAD (Padding): São bytes (0 a 3), de conteúdo zero, adicionados ao campo OP, de forma que o cabeçalho da unidade IP_PDU tenha tamanho múltiplo de quatro bytes.
- INFO (Information): Campo, de tamanho variável, usado para o transporte da unidade de dado do usuário. O tamanho deste campo adicionado ao cabeçalho é limitado a 64 Kbytes.

30. O check de integridade do campo de informação é realizado pela camada CLNAP, através do campo CRC.

4.3.2 Primitivas associadas à camada IP

Este item descreve as primitivas trocadas com a camada LLC para o elemento IWU.

4.3.2.1 Primitivas de interação com a camada LLC

- A. LLC_DATA_request (SA,DA,COS,LLC_SDU): Esta primitiva é usada pela camada IP para requerer a transferência de uma unidade IP_PDU para a camada LLC.
- B. LLC_DATA_indication (LLC_SDU): Esta primitiva é usada para transferência de uma unidade LLC_SDU, recebida sem erros, para a camada IP.

onde,

- LLC_SDU: corresponde a unidade IP_PDU.
- SA e DA: representam os endereços de origem e destino no formato MAC.
- COS (Class of Service): corresponde ao nível de prioridade desejado para o transporte da LLC_PDU na rede local.

4.4 Conclusão

Neste capítulo foi apresentado um detalhamento dos protocolos (ou camadas) dedicados ao serviço não orientado à conexão, ou seja, AAL 4, CLNAP e IP.

As funções da camada AAL 4 (ATM Adaptation Layer) estão implementadas de forma distribuída nas subcamadas SAR (Segmentation and Reassembly), CPCS (Common Part Convergence Sublayer) e SSCS (Service Specific Convergence Sublayer). As funções são as seguintes:

Subcamada SAR:

- Segmentação da unidade CPCS_PDU em unidades SAR_PDU e remontagem destas na recepção.
- Detecção de erro na unidade SAR_PDU.
- Detecção de perda ou de inserção de unidades SAR_PDU.
- Inserção e extração de preenchimento.

Subcamada CPCS

- Verificação da integridade da CPCS_PDU.
- Alocação de memória.

A *subcamada SSCS* é responsável pela adaptação do serviço do usuário à subcamada CPCS. Assim, de acordo com o tipo de serviço requerido à subcamada SSCS, esta pode implementar, por exemplo, funções de segmentação e remontagem, controle de fluxo e recuperação de erros. Para o serviço não orientado à conexão a subcamada SSCS não é implementada, visto que, a camada IP executa as funções de segmentação e remontagem e as demais funções complementares são executadas pelo equipamento do usuário.

O protocolo CLNAP (Connectioless Network Access Protocol) é o responsável pelas principais funções envolvidas no suporte ao serviço não orientado à conexão na B-ISDN, que são:

- Roteamento da unidade CLNAP_PDU.
- Endereçamento de grupo.
- Validação e filtro de endereços.
- Detecção de erro na unidade CLNAP_PDU.
- Inserção e extração de preenchimento.

O protocolo IP (Internet Protocol), implementado somente no elemento IWU, é responsável pela compatibilização entre as redes locais de origem e de destino e destas com a B-ISDN. Esta executa as seguintes funções:

- Remontagem e fragmentação das unidades IP_PDU.
- Conversão de endereços.
- Definição do serviço de transporte.
- Validação da unidade IP_PDU.

Salienta-se que as funções de segmentação e remontagem da subcamada SAR, a subcamada CPCS e as funções de detecção de erro e inserção e extração de preenchimento do protocolo CLNAP, não são implementadas no elemento CLS, devido a opção de operação deste após a recepção da primeira célula da unidade CLNAP_PDU.

Outro aspecto importante a ser observado, é que a especificação do protocolo CLNAP encontra-se em processo de evolução e definição pelo CCITT, de forma que, as funções consideradas necessitarão passar por um processo de revisão e atualização.

Capítulo 5

Especificação e simulação dos elementos de rede

Este capítulo apresenta a especificação funcional dos elementos de rede IWU e CLS, em uma linguagem de descrição formal, e os procedimentos de simulação adotados, considerando um cenário simplificado da B-ISDN para a interconexão de redes locais.

5.1 Especificação formal

A especificação completa de um sistema envolve um conjunto de informações que são expressas de maneira informal, semiformal ou formal, dependendo do tipo de informação, da etapa dentro do ciclo de desenvolvimento do sistema e da metodologia de desenvolvimento utilizada. A linguagem informal (ou natural), por exemplo, pode ser utilizada nas etapas iniciais do desenvolvimento do sistema, para definição dos objetivos e requisitos do sistema, onde detalhes de especificação podem ser omitidos. A linguagem formal, por outro lado, aplica-se de forma mais adequada nas etapas onde se deseja expressar informações de forma precisa.

Em geral, uma linguagem de descrição formal possibilita as seguintes operações:

- a- Especificações não ambíguas, claras e concisas.
- b- Verificação das especificações.
- c- Análise funcional das especificações.
- d- Implementação das especificações.
- e- Teste de conformidade entre a implementação e as especificações.

Para suportar os itens “a”, “d” e “e”, a linguagem de descrição deve possibilitar uma fácil composição, entendimento e expansão das especificações. Isto inclui facilidades, tais como a estruturação do sistema em níveis hierárquicos de abstração, a definição de processos e instâncias e recursividade. Para suportar os itens “b” e “c”, a linguagem deve ter uma forte base matemática de descrição para execução das análises sintática e semântica.

Desde a década de 70, as universidades e as instituições ISO e CCITT vêm atuando em atividades de desenvolvimento e padronização de linguagens de descrição formal. Na década de 80, três linguagens, padronizadas internacionalmente, se destacaram: Estelle [39] e LOTOS [40] definidos pela ISO e SDL [9] definida pelo CCITT.

Para a especificação funcional dos elementos de rede foi adotada a linguagem SDL, motivado por:

- Possibilidade do aproveitamento das especificações constantes das recomendações do CCITT, referenciadas neste trabalho.
- Aproveitamento das especificações, realizadas neste trabalho, na atividade de adição de facilidades B-ISDN, dentro do projeto Trópico, visto que a descrição de partes do sistema em linguagem SDL está, atualmente, inserida na metodologia praticada neste projeto.
- Disponibilidade da ferramenta CASE (Computer Aided Software Engineering) SDT-2 [10], com recursos para simulação do sistema.

5.2 Especificação funcional da B-ISDN

A figura 5.1 apresenta o cenário da B-ISDN adotado para a especificação e simulação dos elementos de rede IWU e CLS.

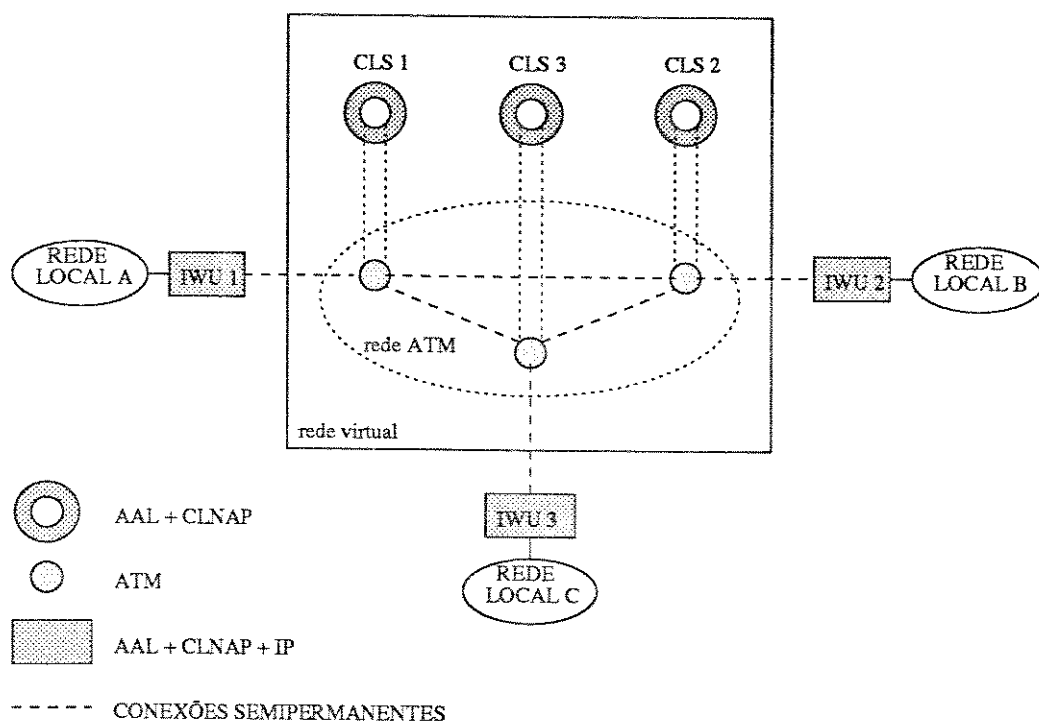


Figura 5.1: Cenário da B-ISDN adotado

Este cenário considera três redes locais ("A", "B" e "C") acessando, através de seus elementos IWU associados, a rede virtual não orientada à conexão, que é constituída de três elementos CLS de acesso interconectados por uma rede de conexões ATM semipermanentes.

Como primeiro nível de abstração, o cenário da B-ISDN adotado será representado por um sistema constituído pelos elementos IWU e pela rede virtual não orientada à conexão. O ambiente do usuário, no caso as redes locais, e as gerências de camada são consideradas como externos ao sistema.

A figura 5.2 representa o sistema “B-ISDN” considerado com seus elementos (ou blocos) e as interfaces de interação entre estes e com o ambiente externo.

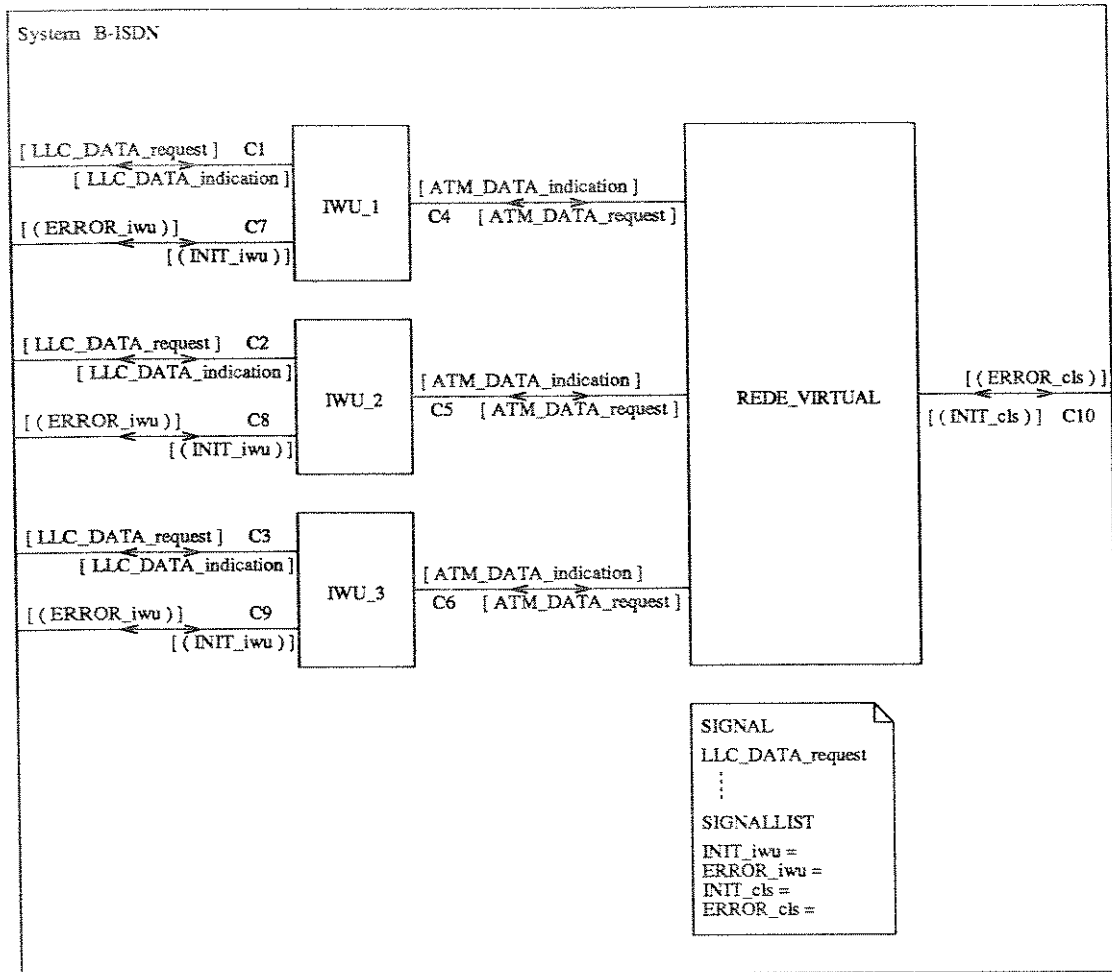


Figura 5.2: Definição do sistema B-ISDN

No sistema “B-ISDN” são identificados os seguintes canais:

- C1, C2 e C3: representam, respectivamente, as interfaces dos elementos IWU_1, IWU_2 e IWU_3 com as redes locais “A”, “B” e “C”. As interações nestas interfaces são realizadas através dos sinais (primitivas) “LLC_DATA_request” e “LLC_DATA_indication”, abstraindo, desta forma, as camadas MAC e FÍSICA das redes locais.
- C4, C5 e C6: representam as conexões semipermanentes que interligam os elementos IWU à rede virtual não orientada à conexão. As interações nestas interfaces ocorrem através das primitivas “ATM_DATA_request” e “ATM_DATA_indication”, abstraindo as camadas ATM e FÍSICA da rede virtual.
- C7, C8, C9 e C10: representam as interfaces do sistema com as gerências de camada. Destas interfaces, as três primeiras são utilizadas para configuração dos elementos

IWU e a última para configuração e monitoração dos componentes da rede virtual não orientada à conexão, que, no caso, são os elementos CLS. Os conjuntos de listas de sinais “INIT_iwu” e “INIT_cls” e “ERROR_iwu” e “ERROR_cls” representam, respectivamente, os sinais de configuração e de indicação de erro dos elementos IWU e CLS.

O bloco “Rede_virtual”, que representa a rede virtual não orientada à conexão, é composto pelos elementos CLS e por uma rede de conexões ATM semipermanentes. A figura 5.3 representa este bloco.

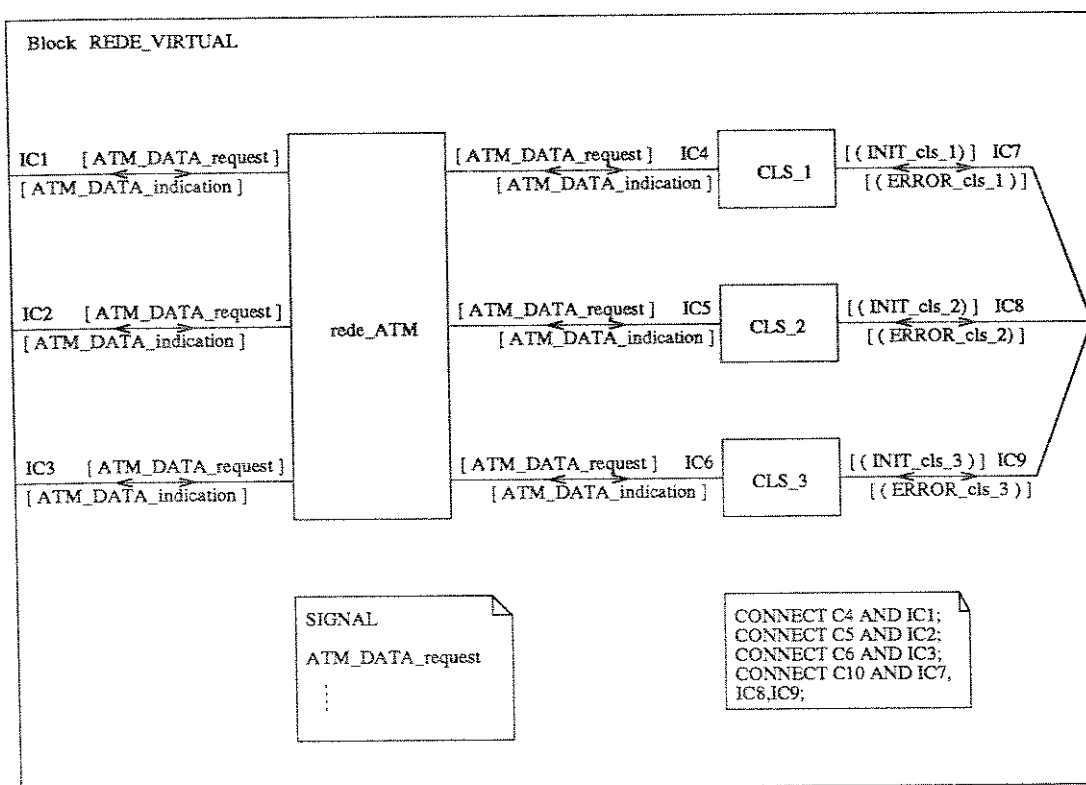


Figura 5.3: Definição do bloco “Rede_virtual”

Os blocos “CLS_1”, “CLS_2” e “CLS_3” representam os elementos CLS de acesso associados, respectivamente, aos blocos “IWU_1”, “IWU_2” e “IWU_3” do sistema “B-ISDN”.

O bloco “rede_ATM” representa a rede de conexões semipermanentes, que recebe um sinal “ATM_DATA_request” de um canal e transmite um sinal “ATM_DATA_indication” para o canal configurado. No cenário da B-ISDN adotado, este bloco está configurado de forma a encaminhar:

- os sinais recebidos pelos canais IC1, IC2 e IC3 para os canais IC4, IC5 e IC6, respectivamente.
- os sinais recebidos pelo canal IC4 para os canais IC1, IC5 ou IC6.
- os sinais recebidos pelo canal IC5 para os canais IC2, IC4 ou IC6.

- os sinais recebidos pelo canal IC6 para os canais IC3, IC4 ou IC5.

Uma vez identificado os blocos funcionais do sistema “B-ISDN” e suas interfaces, as próximas etapas são a especificação funcional e simulação destes blocos isoladamente e, posteriormente a simulação do sistema “B-ISDN” com seus blocos integrados.

A metodologia adotada para a especificação funcional dos blocos considera:

- Especificação dos blocos como sistemas isolados³¹.
- Particionamento visando clareza e flexibilidade.
- Especificação respeitando as restrições da ferramenta SDT-2.
- Descrição de maneira informal³² de funções consideradas complementares à especificação.
- Especificação visando eficiência na simulação e localização de erros.

31. O objeto de simulação para a ferramenta SDT-2 é o sistema.

32. As descrições informais, na linguagem SDL, são declaradas entre aspas simples (ex. 'discard IP_PDU') ou como comentários.

5.3 Especificação funcional do IWU

O elemento IWU, agora designado por sistema “IWU”, foi especificado conforme a figura 5.4. A especificação completa deste sistema é apresentada em [41].

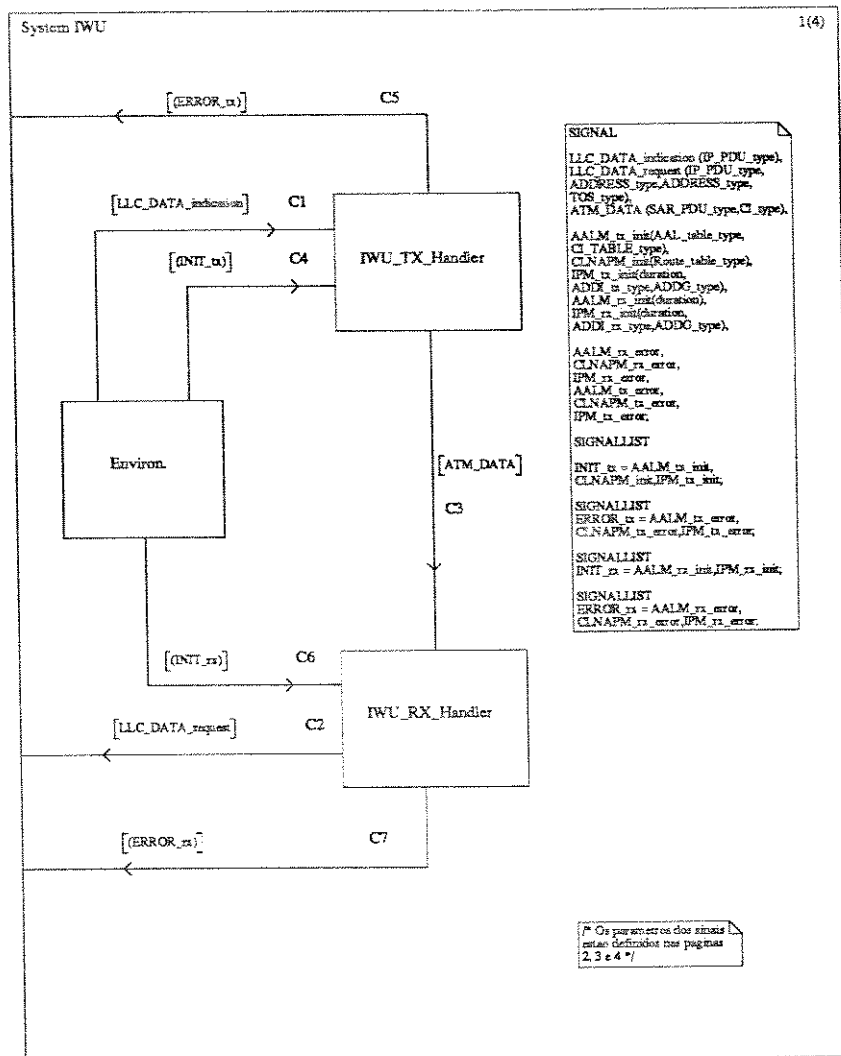


Figura 5.4: Definição do sistema “IWU”

No sistema “IWU”, os blocos “IWU_TX_Handler” e “IWU_RX_Handler” representam, respectivamente, as funções de transmissão e recepção do elemento IWU, que na simulação são considerados como sendo de elementos IWU diferentes, ou seja, local e remoto. Desta forma, o sinal “ATM_DATA” passa a representar a fusão dos sinais “ATM_DATA_request” e “ATM_DATA_indication”.

O bloco “Environ.” foi inserido no sistema para prover a configuração dos demais blocos, através das listas de sinais “INIT_tx” e “INIT_rx” e estimulação do sistema através do sinal “LLC_DATA_indication”.

As listas de sinais “ERROR_tx” e “ERROR_rx” designam, respectivamente, os sinais de erro gerados pelos blocos de transmissão e recepção.

Note que observando o sistema “IWU” no contexto do sistema “B-ISDN” (figura 5.2), as seguintes associações entre canais podem, por exemplo, ser caracterizadas:

sistema "IWU"	sistema "B_ISDN"
C1	C1
C2	C2 ou C3
C3	fusão de C4 com C5 ou C6
C4 e C5	C7
C6 e C7	C8 ou C9

Tabela 5.1: Correlação entre os canais dos sistemas “IWU” e “B-ISDN”

5.3.1 Descrição dos tipos

A seguir são apresentados os tipos dos parâmetros associados aos sinais do sistema “IWU”:

IP_PDU_type: Representa a unidade IP_PDU, na qual o campo FLAG é representado pelo campo MORE e os campos OP e PAD não foram considerados.

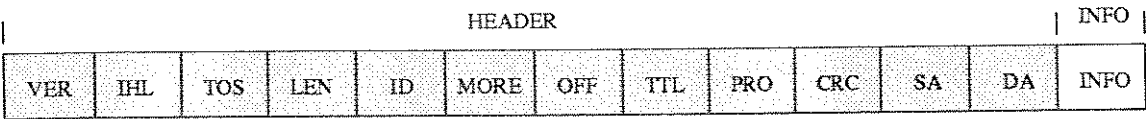


Figura 5.5: IP_PDU_type

ADDRESS_type: Representa os endereços de origem e destino, identificando se estes são individuais ou de grupo. Os endereços nos formatos IP, E.164 e MAC são considerados do mesmo tipo.



Figura 5.6: ADDRESS_type

TOS_type: Representa a classe de qualidade de serviço desejada, expressa através da designação (true ou false) de prioridade, de atraso, de tráfego e de confiabilidade. Os campos TOS (da unidade IP_PDU) e QOS (da unidade CLNAP_PDU) e o parâmetro COS (da primitiva LLC_DATA_request) são considerados do mesmo tipo.

TOS			
priority	delay	throughput	reliability

Figura 5.7: TOS_type

SAR_PDU_type: Representa a unidade SAR_PDU. Para a simulação, este tipo foi definido de forma genérica, representando o conteúdo do campo INFO dos diferentes segmentos da unidade SAR_PDU (BOM, COM, EOM e SSM). Os campos HEADER_IP e INFO correspondem ao tipo IP_PDU_type e o campo TRAILLER_CLNAP representa o campo CRC da unidade CLNAP_PDU.

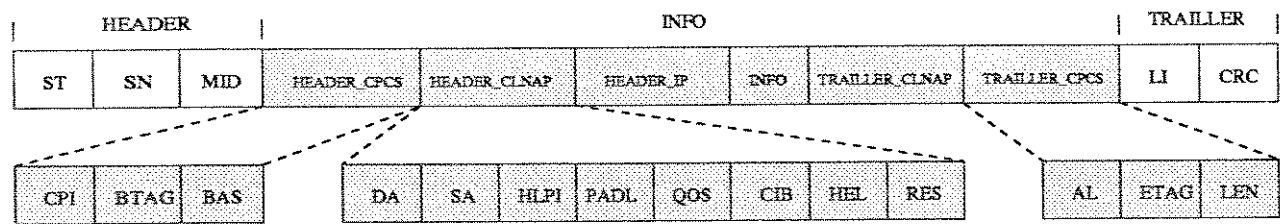


Figura 5.8: SAR_PDU_type

- CI_type:* Identificação das conexões AAL (AAL_CI) e ATM (ATM_CI).
- CI_TABLE_type:* Tabela de associação entre os identificadores das conexões AAL e ATM. Nesta especificação, cada conexão AAL é mapeada em uma única conexão ATM³³.
- AAL_table_type:* Tabela dinâmica que armazena os últimos valores utilizados, na transmissão, para os campos MID da unidade SAR_PDU e TAG (BTAG = ETAG) da unidade CPCS_PDU, por conexão.

ATM_CI	MID	TAG
1	0	0
2	0	0

Tabela 5.2: AAL_table_type (configuração inicial)

Route_table_type: tabela de roteamento que representa a fusão das tabelas 4.3 e 4.4 da camada CLNAP. Esta tabela apresenta as opções de conexões, por endereço de destino e seus atributos de qualidade de serviço.

33. Esta associação ocorre pelo fato da camada AAL não prestar serviços diferenciados por conexão.

- ADDI_tx_type:* Representação da tabela 4.9 que relaciona os endereços IP dos terminais, por rede local, com os endereços E.164 associados.
- ADDI_rx_type:* Representação da tabela 4.8 que relaciona os endereços IP e MAC dos terminais de uma rede local. Para flexibilizar a simulação esta foi expandida para todas as redes locais.
- ADDG_type:* Representação da tabela 4.10 que associa os endereços de grupo IP, E.164 e MAC. Esta tabela, para simplificar a representação, considera que para um dado endereço de grupo IP, os endereços de grupo MAC associados são os mesmos.
- duration:* Intervalo de temporização. Este tipo é usado para definir os tempos máximos para remontagem nos processos “Reassembly”, “IP_tx” e “IP_rx”.

5.3.2 Descrição dos blocos

As figuras 5.9 e 5.10 apresentam, respectivamente, o particionamento do blocos “IWU_TX_Handler” e “IWU_RX_Handler” em processos.

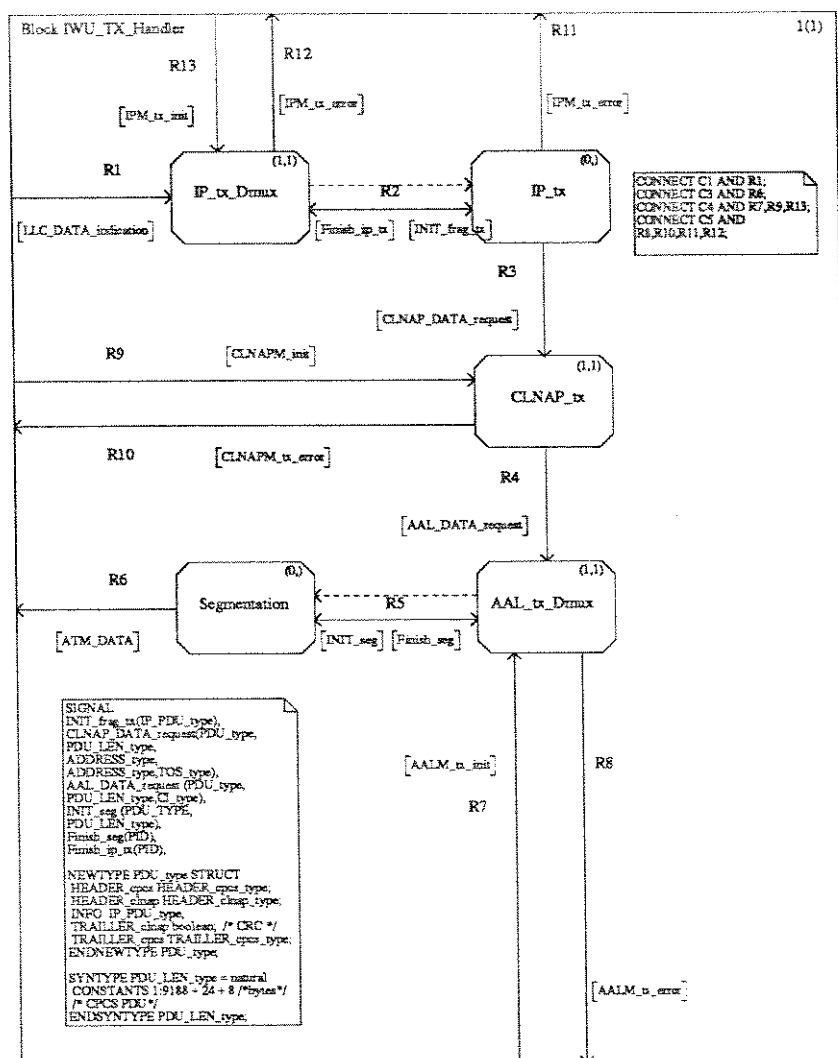


Figura 5.9: Definição do bloco “IWU_TX_Handler”

5.3.2.1 Descrição dos processos

A- Bloco "TWU_TX_Handler":

- "IP_tx_Dmux": Gerencia, aloca e libera as instâncias do processo "IP_tx". A alocação compreende a verificação da disponibilidade de um instância e, no caso positivo, a criação e encaminhamento da unidade IP_PDU para esta. No caso negativo a unidade IP_PDU é descartada.
A liberação da instância é efetuada quando o processo recebe o sinal "Finish_ip_tx" gerado, no término da execução do processo "IP_tx".
Este processo, também, identifica e encaminha os fragmentos de uma IP_PDU para a mesma instância do processo "IP_tx".
- "IP_tx": Executa as funções de remontagem e fragmentação das unidades IP_PDU, tradução de endereços para o formato E.164 e definição da classe de qualidade de serviço para o transporte da unidade IP_PDU através da B-ISDN. A função de remontagem considera a recepção de fragmentos fora de seqüência.
Este processo é temporizado de forma a liberar a instância quando o tempo para remontagem se torna excessivo, como é o caso quando ocorre a perda de um fragmento da unidade IP_PDU.
- "CLNAP_tx": Seleciona a conexão AAL (AAL_CI) para o encaminhamento da unidade CLNAP_PDU. No caso do endereço de destino não constar na tabela de roteamento ou da não disponibilidade de uma conexão com a classe de qualidade de serviço desejada, a unidade CLNAP_PDU é descartada.
Em uma implementação física este processo pode ser implementado como um procedimento dentro do processo "IP_tx", para que suas funções sejam instanciáveis.
- "AAL_tx_Dmux": Gerencia, aloca e libera as instâncias do processo "Segmentation". A alocação compreende a verificação da disponibilidade de um instância e, no caso positivo, a criação e encaminhamento da unidade CPCS_PDU para esta. No caso negativo a unidade CPCS_PDU é descartada.
A liberação da instância é efetuada quando o processo recebe o sinal "Finish_seg", gerado no término da execução do processo "Segmentation".
Este processo, também, controla o conteúdo dos campos BTAG e ETAG da unidade CPCS_PDU e do campo MID da unidade SAR_PDU, para que estes tenham valores seqüenciais para unidades CPCS_PDU transmitidas consecutivamente em uma conexão.
- "Segmentation": Segmenta a unidade CPCS_PDU em unidades SAR_PDU e encaminha estas seqüencialmente para a conexão ATM definida.

B- Bloco "TWU_RX_Handler":

- "IP_rx_Dmux": Gerencia, aloca e libera as instâncias do processo "IP_rx". A alocação compreende a verificação da disponibilidade de um instância e, no caso positivo, a criação e encaminhamento da unidade IP_PDU para esta. No caso negativo a unidade IP_PDU é descartada.
A liberação da instância é efetuada quando o processo recebe o sinal "Finish_ip_rx", gerado no término da execução do processo "IP_rx".

Este processo, também, identifica e encaminha os fragmentos de uma IP_PDU para a mesma instância do processo "IP_rx".

- "IP_rx": Executa as funções de remontagem e fragmentação das unidades IP_PDU, tradução de endereços para o formato MAC e definição da classe de serviço para o transporte da IP_PDU através da rede local. A função de remontagem considera a recepção de fragmentos fora de sequência. Este processo é temporizado de forma a liberar a instância quando o tempo para remontagem se torna excessivo, como, por exemplo, no caso quando ocorre a perda de um fragmento da unidade IP_PDU.
- "CLNAP_rx": Verifica a integridade da unidade CLNAP_PDU descartando as unidades inválidas. Em uma implementação física, este processo pode ser implementado como um procedimento dentro do processo "Reassembly", para que suas funções sejam instanciáveis.
- "AAL_rx_Dmux": Gerencia, aloca e libera as instâncias do processo "Reassembly". A alocação compreende a verificação da disponibilidade de uma instância e, no caso positivo, a criação e encaminhamento da unidade SAR_PDU para esta. No caso negativo a unidade SAR_PDU é descartada. A liberação da instância é efetuada quando o processo recebe o sinal "Finish_reass", gerado no término da execução do processo "Reassembly". Este processo, também, identifica por conexão (ATM_CI) as unidades SAR_PDU que pertencem a mesma unidade CPCS_PDU (campo MID com o mesmo valor) e as encaminha para a mesma instância do processo "Reassembly".
- "Reassembly": Efetua a remontagem das unidades CPCS_PDU a partir das unidades SAR_PDU. Este processo, também, verifica a integridade da unidade CPCS_PDU, descartando as unidades inválidas. Este processo é temporizado de forma a liberar a instância quando o tempo para remontagem se torna excessivo, como é o caso quando ocorre a perda de uma unidade SAR_PDU.

5.3.3 Configuração para simulação

Para a simulação do sistema "IWU" foram considerados o modelo de rede apresentado na figura 5.11 e um conjunto de definições complementares, arbitrariamente selecionado para a análise do comportamento dinâmico do sistema.

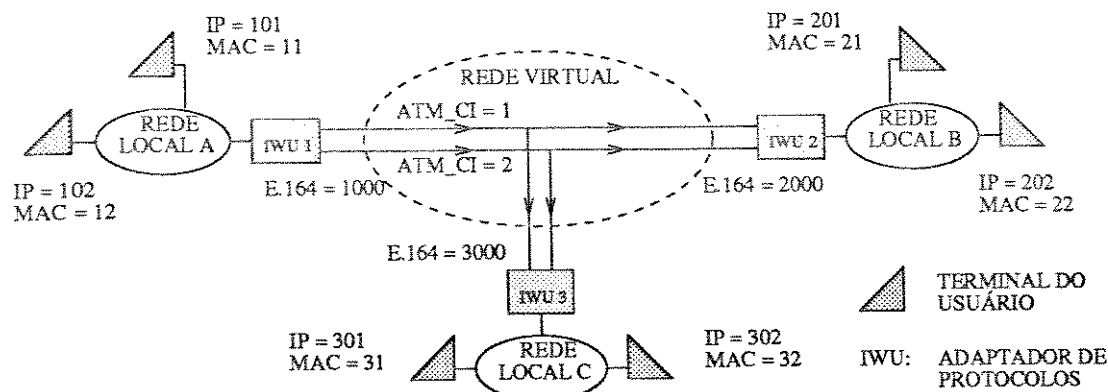


Figura 5.11: Modelo de rede para simulação do sistema “IWU”

Neste modelo foram considerados três redes locais, com dois terminais cada, sendo que, a rede local “A” é a geradora de unidades de dado e as redes locais “B” e “C” as receptoras. Considera-se, também, a existência de duas conexões alternativas, com mesma classe de qualidade de serviço, entre os elementos IWU e rede virtual (CLS de acesso).

A partir deste modelo, o sistema “IWU” foi configurado com as seguintes tabelas:

AAL_CI	ATM_CI
1	1
2	2

Tabela 5.3: Associação entre as conexões AAL e ATM (“CI_TABLE_type”)

DA (E.164)	AAL_CI	CLASSE DE QUALIDADE DE SERVIÇO (QOS)			
		priority	delay	throughput	reliability
2000	1	false	false	false	false
	2	false	false	false	false
3000	1	false	false	false	false
	2	false	false	false	false

Tabela 5.4: Opções de roteamento (“Route_table_type”)

ENDEREÇO IP	ENDEREÇO E.164
101	1000
102	
201	2000
202	
301	3000
302	

Tabela 5.5: Associação entre os endereços E.164 e IP por rede local (“ADDI_tx_type”)

ENDEREÇO IP	ENDEREÇO MAC
101	11
102	12
201	21
202	22
301	31
302	32

Tabela 5.6: Associação entre os endereços IP e MAC das redes locais (“ADDI_rx_type”)

Além destas tabelas, as seguintes definições foram consideradas:

- Tempos máximos para remontagem dos processos “IP_tx”, “IP_rx” e “Reassembly” de 20, 20 e 10 unidades de tempo³⁴.
- Número máximo de instâncias para os processos “IP_tx”, “IP_rx”, “Segmentation” e “Reassembly” igual a seis³⁵.
- Tamanho máximo para a unidade IP_PDU, suportado pelas redes locais “B” e “C” igual ao tamanho máximo suportado pela B-ISDN, ou seja, igual a 9188 bytes³⁶.
- Dois endereços de grupo, onde o primeiro (666) representa os terminais com endereços IP iguais a 101 e 201, e o segundo (777) representa terminais com endereços de 101 e 301. A tabela 5.7 apresenta os endereços de grupo considerados.

ENDEREÇO IP_g	ENDEREÇO E.164_g	ENDEREÇO MAC_g
666	6666	66
777	7777	77

E.164_g: 6666 = { 1000,2000 }

E.164_g: 7777 = { 1000,3000 }

MAC_g a: 66 = {11}

MAC_g a: 77 = {11}

MAC_g b: 66 = {21}

MAC_g b: 77 = {31}

Tabela 5.7: Definição dos endereços de grupo (“ADDG_type”)

34. Esta unidade, definida pelo simulador, refere-se ao intervalo de tempo de execução de uma transição entre estados de um processo.

35. O objetivo deste número é limitar a capacidade de tratamento destes processos.

36. Note que este valor satisfaz a condição de que o campo de informação da unidade IP_PDU seja múltiplo de 8 bytes. Assim, considerando que o cabeçalho da IP_PDU é de 20 bytes, tem-se que: $9188 = 20 + 8 * ((9188 - 20) / 8)$.

5.3.3.1 Procedimento

A validação do sistema foi realizada por etapas, estimulando o sistema de forma a envolver gradualmente as funções implementadas.

A estimulação é feita através dos sinais “LLC_DATA_indication”, variando o número deste sinais, ou seja, das unidades IP_PDU transmitidas e o conteúdo dos campos destas unidades. A verificação é realizada através da análise do percurso da unidade IP_PDU até a geração do sinal “LLC_DATA_request”.

Os seguintes estímulos foram gerados:

- 1- Uma IP_PDU (SA = 101, DA = 201, LEN = 100 bytes, ID = 1, TOS = {false, false, false, false}, VER = true, IHL = 20 bytes, PRO = true, CRC = true, MORE = false, OFF = 0, TTL = 3)

Percurso da unidade IP_PDU através dos processos:

- | | |
|-----------------|---|
| “IP_tx_Dmux”: | alocação de instância. |
| “IP_tx”: | tradução de endereços, mapeamento da classe de qualidade de serviço e verificação de erros. |
| “CLNAP_tx”: | seleção da conexão. |
| “AAL_tx_Dmux”: | alocação de instância. |
| “Segmentation”: | geração de unidades SAR_PDU. |
| “AAL_rx_Dmux”: | alocação de instâncias. |
| “Reassembly”: | remontagem das unidades SAR_PDU e verificação de erros. |
| “CLNAP_rx”: | verificação de erros. |
| “IP_rx_Dmux”: | alocação de instância. |
| “IP_rx”: | tradução de endereços, mapeamento da classe de qualidade de serviço e verificação de erros. |
-
- 2- Alteração do campo LEN para 400 bytes e análise dos processos “Segmentation” e “Reassembly”.
 - 3- Alteração dos campos DA (301, 401, 666³⁷) e TOS ({true, false, false, false}) e análise dos processos “IP_tx” e “CLNAP_tx”. Campo LEN = 100 bytes.
 - 4- Vinte unidades IP_PDU, com campos ID de 1 a 20, para análise dos processos “IP_tx_Dmux”, “AAL_tx_Dmux”, “AAL_rx_Dmux” e “IP_rx_Dmux” e do comportamento dinâmico do sistema.
 - 5- Uma unidade IP_PDU com campo LEN = 20000 bytes para verificação dos processos “IP_tx” (fragmentação) e “IP_rx” (remontagem e fragmentação).
 - 6- Vinte unidades IP_PDU, com campos ID de 1 a 20 e campos LEN = 20000 bytes, para verificação do comportamento dinâmico do sistema.

37. Endereço de grupo.

5.4 Especificação funcional do CLS

O elemento CLS, designado por sistema “CLS”, foi especificado conforme a figura 5.12. A especificação completa deste sistema é apresentada em [42].

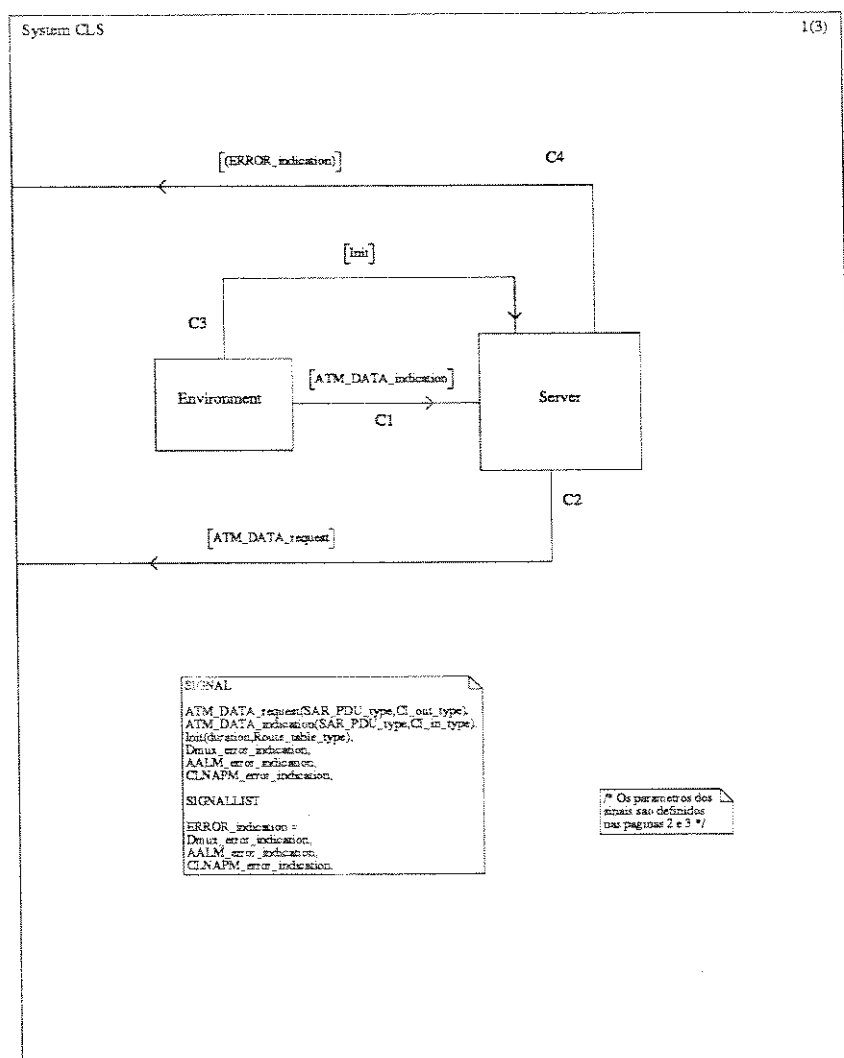


Figura 5.12: Definição do sistema “CLS”

Neste sistema o bloco “Server” representa as funções do elemento CLS, recebendo os sinais “ATM_DATA_indication” e os encaminhando para o destino através dos sinais “ATM_DATA_request”.

O bloco “Environment” foi inserido no sistema para prover a configuração do bloco “Server”, através do sinal “Init” e estimulação do sistema, através do sinal “ATM_DATA_indication”.

A lista de sinal “ERROR_indication” representa os sinais de erro gerados pelos processos do bloco “Server”.

Note que as seguintes correlações entre os canais do sistema “CLS” e do bloco “REDE_VIRTUAL” podem ser efetuadas:

- C1 e C2 com IC4, IC5 ou IC6
- C3 e C4 com IC7, IC8 ou IC9

5.4.1 Descrição dos tipos

A seguir são apresentados os tipos dos parâmetros associados aos sinais do sistema “CLS”:

SAR_PDU_type: Representa a unidade SAR_PDU. Para a simulação este tipo foi definido de forma genérica para os diferentes segmentos da unidade SAR_PDU (BOM, COM, EOM e SSM), e considerando apenas os campos relevantes à operação do sistema “CLS”. Os campos SA e DA são do tipo “ADDRESS_type” e o campo QOS do tipo “TOS_type”.

HEADER			INFO			TRAILLER	
ST	SN	MID	DA	SA	QOS	LI	CRC

Figura 5.13: SAR_PDU_type

CI_in_type: Identificação da conexão ATM (ATM_CI) de entrada.

CI_out_type: Identificação da conexão ATM (ATM_CI) de saída.

Route_table_type: Representa a tabela de roteamento.

duration: Intervalo de temporização. Este tipo é usado para definir o tempo máximo de execução para o processo “Router”.

5.4.2 Descrição dos blocos

A figura 5.14 apresenta o particionamento do bloco “Server” em processos.

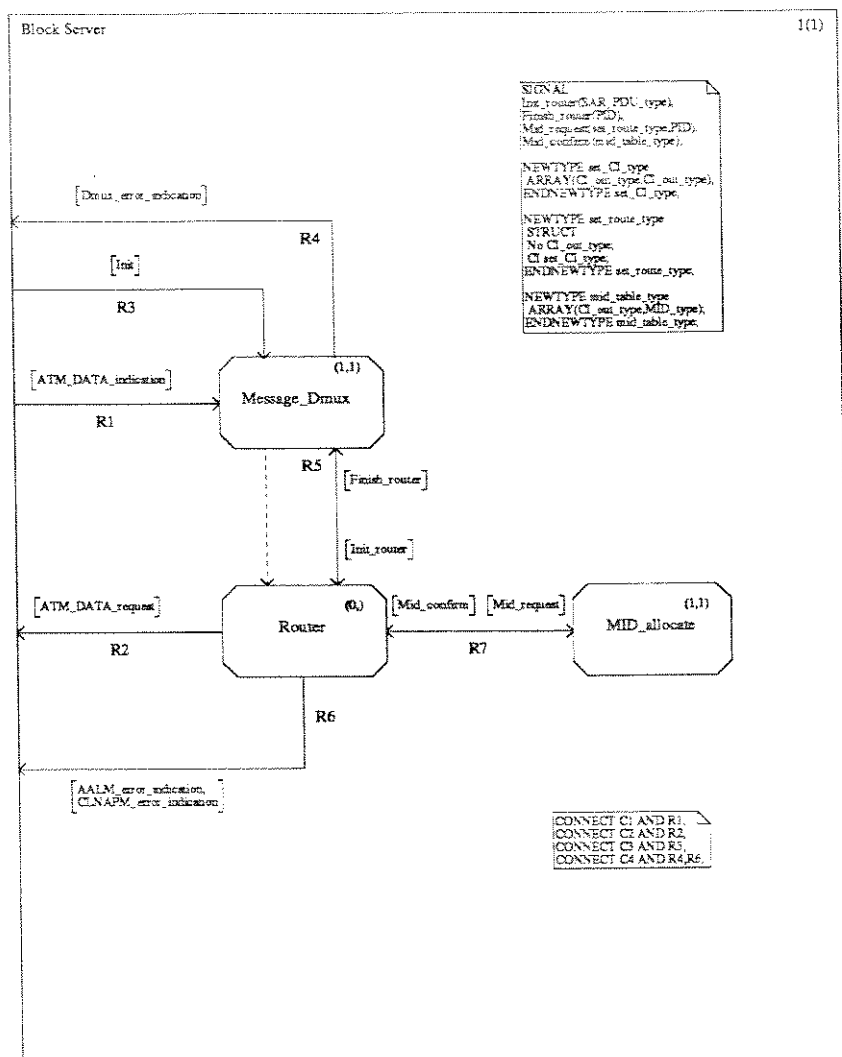


Figura 5.14: Definição do bloco “Server”

O particionamento do bloco “Server” foi realizado de forma a concentrar as funções da camada CLNAP e as principais funções da camada AAL 4, do elemento CLS, no processo “Router” e atribuindo as funções complementares da camada AAL 4 aos processos “Message_Dmux” e “MID_allocate”.

O processo “Router” foi definido como instanciável, sob controle do processo “Message_Dmux”, de forma que este último ao receber as unidades SAR_PDU, pertencentes à unidades CPCS_PDU distintas, as distribui para diferentes instâncias do processo “Router”, para que sejam roteadas paralelamente.

5.4.2.1 Descrição dos processos

- “Message_Dmux”: Gerencia, aloca e libera as instâncias do processo “Router. A alocação compreende a verificação da disponibilidade de um instância e, no caso positivo, a criação e encaminhamento da unidade SAR_PDU para esta. No caso negativo a unidade SAR_PDU é descartada.
A liberação da instância é efetuada quando o processo recebe o sinal “Finish_router”, gerado no término da execução do processo “Router”.
Este processo, também, identifica por conexão (ATM_CI) as unidades SAR_PDU que pertencem a mesma unidade CPCS_PDU (campo MID com o mesmo valor) e as encaminha para a mesma instância do processo “Router”.
- “Router”: Identifica a primeira unidade SAR_PDU pertencente a unidade CPCS_PDU e, a partir das informações de endereço de destino e da classe de qualidade de serviço presentes nesta, seleciona as conexões ATM (ATM_CI) para as quais será encaminhada³⁸. As demais unidades SAR_PDU, da mesma CPCS_PDU, são encaminhadas para as mesmas conexões.
No caso do endereço de destino não constar na tabela de roteamento, da não disponibilidade de uma conexão com a classe de qualidade de serviço desejada ou da detecção de uma SAR_PDU inválida, a unidade SAR_PDU e as demais unidades consecutivas, da mesma unidade CPCS_PDU, são descartadas.
Este processo é temporizado de forma a liberar a instância quando o tempo para o roteamento de todas unidades SAR_PDU, da mesma CPCS_PDU, se torna excessivo, como por exemplo, quando ocorre a perda de uma unidade SAR_PDU.
As funções de seleção de conexão, baseada em endereço de grupo, de validação e de filtro de endereços, não foram detalhadas, aparecendo referenciadas neste processo de maneira informal.
- “MID_allocate”: Define, para cada conexão selecionada pelas instâncias do processo “Router”, o valor a ser atribuído ao campo MID das unidades SAR_PDU da mesma CPCS_PDU. Estes valores são atribuídos de forma sequencial, por conexão, às unidades SAR_PDU de unidades CPCS_PDU distintas.
Esta função foi centralizada, em uma única instância, para garantir que unidades SAR_PDU de unidades CPCS_PDU consecutivas, em uma mesma conexão, possuam o campo MID com valores distintos.

5.4.3 Configuração para simulação

Para a simulação do sistema “CLS” foram considerados o modelo de rede apresentado na figura 5.15 e um conjunto de definições complementares, arbitrariamente selecionado para a análise do comportamento dinâmico do sistema.

38. No caso de endereçamento de grupo mais de uma conexão pode ser selecionada.

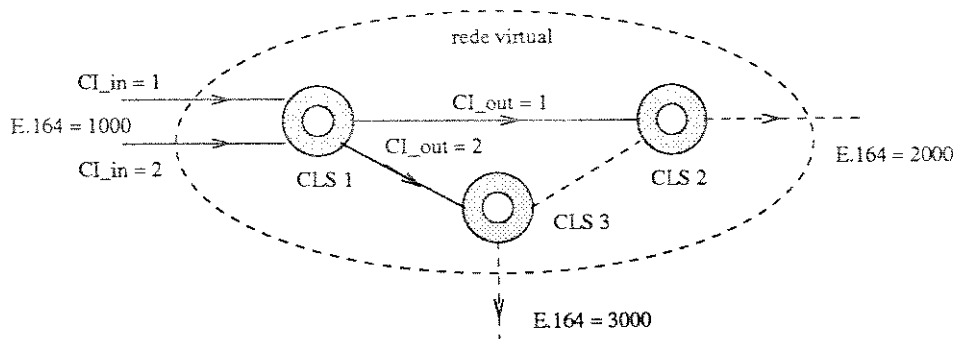


Figura 5.15: Modelo de rede para simulação do sistema “CLS”

Neste modelo o elemento CLS (de acesso) foi configurado com duas conexões ATM (ATM_CI) de entrada e duas conexões de saída, com classes de qualidade de serviço diferentes.

As opções de roteamento, para este modelo, são apresentadas na tabela 5.8.

DA (E.164)	ATM_CI	CLASSE DE QUALIDADE DE SERVIÇO (QOS)			
		priority	delay	throughput	reliability
2000	1	false	false	false	false
	2	true	true	true	true
3000	1	false	false	false	false
	2	true	true	true	true

Tabela 5.8: Opções de roteamento (“Route_table_type”)

Além desta tabela, as seguintes definições foram consideradas:

- Tempo máximo para roteamento de todas unidades SAR_PDU de uma CPCS_PDU de 20 unidades de tempo.
- Número máximo de instâncias para o processo “Router” igual a seis.
- Seleção das conexões 1 e 2, no caso de endereçamento de grupo.

5.4.3.1 Procedimento

A validação do sistema foi realizada por etapas, estimulando o sistema de forma a envolver gradualmente as funções implementadas.

A estimulação do sistema é feita através dos sinais “ATM_DATA_indication”, variando o número destes sinais e os seus parâmetros que são a unidade “SAR_PDU” e a

conexão de entrada “CI_in”. A verificação é realizada através da análise do percurso das unidades SAR_PDU até a geração do sinal “ATM_DATA_request”.

Os seguintes estímulos foram gerados:

- 1- Três unidades SAR_PDU, de uma mesma CPCS_PDU, através da conexão CI_in = 1.
 - (ST = BOM, SN = 1, MID = 1, SA = 1000, DA = 2000, QOS = {false, false, false, false}, LI = 44, CRC = true)
 - (ST = COM, SN = 2, MID = 1, LI = 44, CRC = true)
 - (ST = EOM, SN = 3, MID = 1, LI = 44, CRC = true)

Processos estimulados:

“Message_Dmux”: alocação de instância.

“Router”: verificação da integridade da unidade SAR_PDU e seleção da conexão de saída.

“MID_allocate”: definição do campo MID da unidade SAR_PDU.

- 2- Vinte unidades SAR_PDU (SN = 1 a 20), de uma mesma CPCS_PDU, para análise do processo “Router”.
- 3- Alteração dos campos DA (3000, 4000) e QOS ({true, true, true, true} e {true, false, false, false}), para análise do processo “Router”.
- 4- Dez conjuntos de três unidades SAR_PDU (SN = 1 a 3), de unidades CPCS_PDU diferentes (MID = 1 a 10), para análise dos processos “Message_Dmux” e “MID_allocate” e do comportamento dinâmico do sistema.
- 5- Dez conjuntos de três unidades SAR_PDU (SN = 1 a 3), de unidades CPCS_PDU diferentes (MID = 1 a 10), alternando o conteúdo do campo QOS ({true, true, true, true} e {false, false, false, false}), para análise do processo “MID_allocate” e do comportamento dinâmico do sistema.
- 6- Dez conjuntos de três unidades SAR_PDU (SN = 1 a 3), de unidades CPCS_PDU diferentes (MID = 1 a 10), alternando o conteúdo do campo QOS ({true, true, true, true} e {false, false, false, false}), para as duas conexões de entrada, para análise dos processos “Message_Dmux” e “Router”, e do comportamento dinâmico do sistema.

5.5 Estudo de caso

Esta seção apresenta um exemplo da sistemática de avaliação das especificações realizada para o sistema “CLS”, considerando a configuração apresentada no item 5.4.3.

A sistemática considerada está dividida em três etapas: definição dos estímulos que exercitarão o sistema; análise do comportamento esperado do sistema, perante os estímulos definidos; estimulação do sistema e comparação do comportamento do sistema observado com o esperado.

- A- Conforme apresentado na figura 5.12, o sistema “CLS” é estimulado através de sinais “ATM_DATA_indication”, compostos de dois parâmetros dos tipos “SAR_PDU_type” e “CI_in_type” (veja item 5.4.1), que são gerados pelo bloco “Environment”. Neste exemplo, foram definidos para a estimulação do sistema 34 sinais “ATM_DATA_indication”, com os parâmetros “SAR_PDU”, correspondentes a cinco unidades CPCS_PDU³⁹ de 132 bytes (3*44) e a uma unidade CPCS_PDU de 836 bytes (19*44)⁴⁰. Os parâmetros “CI_in”, que indicam a conexão de entrada para as unidades SAR_PDU, foram definidas com os valores “1” ou “2”. A figura 5.16 ilustra os estímulos definidos acima.

39. As unidades CPCS_PDU são segmentadas em unidades SAR_PDU, para serem transportadas no campo de informação da célula ATM.

40. Esta CPCS_PDU foi gerada para análise do processo de verificação da integridade da sequência das unidades SAR_PDU, quando o número destas é maior que o limite definido para o campo SN (16).

SAR_PDU													CI_in	
ST	SN	MID	DA		SA		QOS				LI	CRC	ATM_CI	
			GROUP	ADDRESS	GROUP	ADDRESS	priority	delay	throughp	reliabil				
CABEÇALHO			INFO									TRAILER		
CPCS PDU 1	BOM	0	1	FALSE	2000	FALSE	1000	FALSE	FALSE	FALSE	FALSE	44	TRUE	1
	COM	1	1									44	TRUE	1
	EOM	2	1									44	TRUE	1
CPCS PDU 2	BOM	0	2	FALSE	3000	FALSE	1000	TRUE	TRUE	TRUE	TRUE	44	TRUE	2
	COM	1	2									44	TRUE	2
	EOM	2	2									44	TRUE	2
CPCS PDU 3	BOM	0	3	FALSE	1000	FALSE	1000	FALSE	FALSE	FALSE	FALSE	44	TRUE	1
	COM	1	3									44	TRUE	1
	EOM	2	3									44	TRUE	1
CPCS PDU 4	BOM	0	4	FALSE	2000	FALSE	1000	TRUE	TRUE	TRUE	TRUE	44	TRUE	2
	COM	1	4									44	TRUE	2
	EOM	2	4									44	TRUE	2
CPCS PDU 5	BOM	0	5	FALSE	3000	FALSE	1000	FALSE	FALSE	FALSE	FALSE	44	TRUE	1
	COM	1	5									44	TRUE	1
	EOM	2	5									44	TRUE	1
CPCS PDU 6	BOM	0	6	FALSE	2000	FALSE	1000	FALSE	FALSE	FALSE	FALSE	44	TRUE	1
	COM	1	6									44	TRUE	1
	COM	2	6									44	TRUE	1
	COM	15	6									44	TRUE	1
	COM	0	6									44	TRUE	1
	COM	1	6									44	TRUE	1
	EOM	2	6									44	TRUE	1

Figura 5.16: Parâmetros “SAR_PDU” e “CI_in” definidos

Observe, também, na figura 5.16 que:

- Os campos MID das unidades SAR_PDU, das seis unidades CPCS_PDU, assumem valores de 1 a 6.

- Os campos INFO das unidades SAR_PDU são definidos apenas para as primeiras unidades SAR_PDU das unidades CPCS_PDU.
- Os endereços de destino (campo DA) e origem (campo SA) da unidade CPCS_PDU 3 são iguais (1000).
- A classe de qualidade de serviço (campo QOS) não é a mesma para as diferentes unidades CPCS_PDU.

B- Espera-se as seguintes atuações dos processos:

“Message_Dmux”: alocação de instâncias do processo “Router”, de 1 a 6, para tratamento das seis unidades CPCS_PDU, pois o número máximo de instâncias para este processos é igual a seis.

“Router”: definição da conexão de saída (CI_out) de acordo com a tabela 5.8, de roteamento. No caso espera-se, por exemplo, que as unidades SAR_PDU da CPCS_PDU 2 sejam encaminhadas para a conexão 2 e que as unidades SAR_PDU da CPCS_PDU 3 sejam descartadas, visto que, o endereço de destino (DA = 1000) não consta da tabela de roteamento.

“MID_allocate”: alocação seqüencial, para cada conexão de saída, de valores para os campos MID das unidades SAR_PDU.

C- A verificação do comportamento do sistema foi realizada, através do simulador, configurando este para armazenar os “traces” de simulação em arquivo e aplicando os seguintes comandos:

- “Set-Trace – 1”: para restringir os “traces” de simulação aos sinais, e seus parâmetros, da interface do sistema com o ambiente externo, ou seja, os sinais “ATM_DATA_request”, “Dmux_error_indication”, “AALM_error_indication” e “CLNAPM_error_indication”.
- “Go-Forever”: para ativação e execução dos processos. A estimulação do sistema é efetuada com a ativação do processo interno ao bloco “Environment”.

A figura 5.17 representa⁴¹ os parâmetros dos sinais “ATM_DATA_request” extraídos do “trace” gerado pelo simulador.

41. Esta representação visa facilitar a interpretação dos resultados obtidos através do simulador.

SAR_PDU														CI_out
ST	SN	MID	DA		SA		QOS				LI	CRC	ATM_CI	
			GROUP	ADDRESS	GROUP	ADDRESS	priority	delay	throughput	reliability				
CABEÇALHO			INFO									TRAILER		
CPCS PDU 1	BOM	0	1	FALSE	2000	FALSE	1000	FALSE	FALSE	FALSE	FALSE	44	TRUE	1
	COM	1	1	FALSE	0	FALSE	0	FALSE	FALSE	FALSE	FALSE	44	TRUE	1
	EOM	2	1	FALSE	0	FALSE	0	FALSE	FALSE	FALSE	FALSE	44	TRUE	1
CPCS PDU 2	BOM	0	1	FALSE	3000	FALSE	1000	TRUE	TRUE	TRUE	TRUE	44	TRUE	2
	COM	1	1	FALSE	0	FALSE	0	FALSE	FALSE	FALSE	FALSE	44	TRUE	2
	EOM	2	1	FALSE	0	FALSE	0	FALSE	FALSE	FALSE	FALSE	44	TRUE	2
CPCS PDU 4	BOM	0	2	FALSE	2000	FALSE	1000	TRUE	TRUE	TRUE	TRUE	44	TRUE	2
	COM	1	2	FALSE	0	FALSE	0	FALSE	FALSE	FALSE	FALSE	44	TRUE	2
	EOM	2	2	FALSE	0	FALSE	0	FALSE	FALSE	FALSE	FALSE	44	TRUE	2
CPCS PDU 5	BOM	0	2	FALSE	3000	FALSE	1000	FALSE	FALSE	FALSE	FALSE	44	TRUE	1
	COM	1	2	FALSE	0	FALSE	0	FALSE	FALSE	FALSE	FALSE	44	TRUE	1
	EOM	2	2	FALSE	0	FALSE	0	FALSE	FALSE	FALSE	FALSE	44	TRUE	1
CPCS PDU 6	BOM	0	3	FALSE	2000	FALSE	1000	FALSE	FALSE	FALSE	FALSE	44	TRUE	1
	COM	1	3	FALSE	0	FALSE	0	FALSE	FALSE	FALSE	FALSE	44	TRUE	1
	COM	2	3	FALSE	0	FALSE	0	FALSE	FALSE	FALSE	FALSE	44	TRUE	1

	COM	15	3	FALSE	0	FALSE	0	FALSE	FALSE	FALSE	FALSE	44	TRUE	1
	COM	0	3	FALSE	0	FALSE	0	FALSE	FALSE	FALSE	FALSE	44	TRUE	1
COM	1	3	FALSE	0	FALSE	0	FALSE	FALSE	FALSE	FALSE	44	TRUE	1	
EOM	2	3	FALSE	0	FALSE	0	FALSE	FALSE	FALSE	FALSE	44	TRUE	1	

Figura 5.17: Parâmetros “SAR_PDU” e “CI_out” observados

Note, na figura 5.17 que:

- Para os campos INFO, das unidades SAR_PDU, não definidos na figura 5.16 , o simulador atribui: “0” aos campos do tipo *natural* e “FALSE” aos campos do tipo *boolean*.
- A unidade CPCS_PDU 3 não está presente.
- Os parâmetros “CI_out” estão de acordo com a tabela 5.8.
- Os campos MID das unidades SAR_PDU assumem valores sequenciais por conexão de saída.

A figura 5.18 ilustra uma parte do “trace” de simulação no formato apresentado pelo simulador.

```

:
* OUTPUT of ATM_DATA_request from Router: 2 to env: 1
* Parameter(s): ( ( BOM, 0, 1 ), ( ( false, 3000 ), ( false, 1000 ),
( true, true, true, true ) ) ), ( 44, true ) ), 2
* OUTPUT of ATM_DATA_request from Router: 2 to env: 1
* Parameter(s): ( ( COM, 1, 1 ), ( ( false, 0 ), ( false, 0 ),
( false, false, false, false ) ) ), ( 44, true ) ), 2
* OUTPUT of CLNAPM_error_indication from Router: 3 to env: 1
* OUTPUT of ATM_DATA_request from Router: 2 to env: 1
* Parameter(s): ( ( EOM, 2, 1 ), ( ( false, 0 ), ( false, 0 ),
( false, false, false, false ) ) ), ( 44, true ) ), 2
* OUTPUT of AALM_error_indication from Router: 3 to env: 1
* OUTPUT of AALM_error_indication from Router: 3 to env: 1
:

```

Figura 5.18: Parte do “trace” de simulação

Note que este “trace” apresenta:

- os sinais “ATM_DATA_request” referentes à unidade CPCS_PDU 2, gerados pela instância “2” do processo “Router”.
- o sinal “CLNAPM_error_indication”, que indica a falha de roteamento e descarte da primeira unidade SAR_PDU da CPCS_PDU 3, e os sinais “AALM_error_indication”, que indicam o descarte das demais unidades SAR_PDU desta CPCS_PDU. Estes sinais são gerados pela instância “3” do processo “Router”.

5.6 Conclusão

Este capítulo apresentou a especificação funcional, em linguagem SDL, dos elementos de rede IWU e CLS e os procedimentos de simulação adotados, considerando um cenário de rede simplificado que permitisse uma análise abrangente das funcionalidades.

Foi verificado que a linguagem SDL (Specification and Description Language) é bastante apropriada para a especificação funcional dos elementos de rede considerados, devido a facilidade de descrição em diferentes níveis hierárquicos de abstração, a flexibilidade dos tipos abstratos de dados e aos recursos para tratamento de eventos temporais.

Quanto a ferramenta CASE SDT-2 utilizada, foram constatadas as facilidades de manuseio e de localização e correção de falhas na especificação e a disponibilidade de uma variedade de opções de comandos do simulador, que facilitam a análise do comportamento dinâmico do sistema e a simulação de condições de exceção.

Foi apresentado, também, um estudo de caso que ilustra a sistemática utilizada no processo de verificação das especificações dos elementos IWU e CLS.

Capítulo 6

Conclusão

Este trabalho abordou um tema que representa a concatenação de duas tendências observadas em nível mundial. A primeira reflete a evolução da tecnologia de comutação segundo a técnica ATM (Asynchronous Transfer Mode) e a segunda aponta uma forte demanda de mercado para a interconexão de redes locais e metropolitanas que operam com altas taxas de transmissão e protocolos não orientados à conexão.

No sentido de contemplar o assunto de forma abrangente e didática, foram apresentados os conceitos básicos da B-ISDN e da técnica de transporte ATM, aspectos de arquitetura de rede, as opções de implementação dos elementos de rede envolvidos, os protocolos associados e suas funções, as especificações funcionais e as metodologias de desenvolvimento e simulação utilizadas.

Além disto, no desenvolvimento deste trabalho foi utilizada uma ampla gama de conceitos complementares relativos ao modelamento do sistema em camadas, a protocolos, à descrição e particionamento do sistema na linguagem SDL (Specification and Description Language) e automação do processo de desenvolvimento de sistemas através da ferramenta CASE (Computer Aided Software Engineering) SDT-2 (SDL Design Tool).

Desta forma, o conteúdo deste trabalho é um subsídio importante para a capacitação das empresas de telecomunicações, tais como, as operadoras do sistema Telebrás, para a execução de planejamentos estratégicos de evolução de rede e elaboração de licitações e para o Centro de Pesquisa e Desenvolvimento da Telebrás e empresas privadas, para o desenvolvimento de equipamentos roteadores e servidores.

A utilização da linguagem SDL mostrou-se bastante apropriada para a especificação funcional dos elementos de rede, devido a facilidade de descrição em diferentes níveis hierárquicos de abstração, a flexibilidade dos tipos abstratos de dados e aos recursos para tratamento de eventos temporais.

Quanto a ferramenta CASE utilizada, foram constatadas as facilidades de manuseio e de localização e correção de falhas na especificação e a disponibilidade de uma variedade de opções de comandos do simulador, que facilitam a análise do comportamento dinâmico do sistema e a simulação de condições de exceção.

Para a transformação das especificações funcionais dos elementos IWU e CLS apresentadas em produtos, são necessárias, ainda, as seguintes etapas:

A- Complementação das especificações funcionais:

- Expansão das funções que aparecem referenciadas de maneira informal, ou como comentários.

- Especificação das funções de operação, manutenção e supervisão relativas às gerências de camada e ao plano de gerência.
 - Especificação de um mecanismo de controle de fluxo e alocação de banda, para o elemento IWU.
- B- Simulação das especificações apresentadas no item “A”.
- C- Definição da plataforma de hardware e do sistema operacional.
- D- Mapeamento das especificações funcionais na plataforma definida no item “C”.
- E- Modelamento e análise de desempenho dos elementos de rede.
- F- Geração de código e projeto do hardware. Nesta etapa podem ser utilizados os aplicativos para geração de código e bibliotecas da ferramenta SDT-2. Note que a ferramenta SDT-2 não contempla o projeto do hardware, devendo este ser realizado em um ambiente de desenvolvimento específico.

Outro aspecto importante a ser considerado na análise deste trabalho é que as especificações funcionais realizadas foram baseadas em recomendações do CCITT que estão em processo contínuo de evolução e definição, de forma que estas especificações necessitarão passar por um processo de revisão e atualização.

Apêndice A

Modelo em camadas

A.1 Conceitos básicos

De acordo com o modelo de referência de interconexão de sistemas abertos da ISO, [43] e [20], um sistema aberto é representado, logicamente, por um conjunto de sub-sistemas organizados sequencialmente, na vertical, em estruturas denominadas por camadas.

Cada camada é constituída por um conjunto de entidades, que são elementos ativos que executam as funções da camada. As entidades pertencentes à mesma camada são denominadas entidades pares.

As entidades pares de uma camada, de sistemas diferentes, se comunicam através dos serviços prestados pela camada imediatamente abaixo e assim sucessivamente. Desta forma, o serviço prestado pela camada “N” é realizado pela execução das funções das camadas “N”, “N-1”, “N-2”, etc. O acesso através do qual a camada “N” presta serviço para a camada “N+1”, é denominado por N-SAP (Service Access Point), que é identificado através do N-SAPI (Service Access Point Identifier).

A associação estabelecida, pela camada “N”, entre duas ou mais entidades da camada “N+1”, para a transferência de dados é denominada por conexão da camada “N”. A terminação de uma conexão da camada “N” em um N-SAP, é designada por N-CE (Connection Endpoint) e identificada através do N-CEI (Connection Endpoint Identifier).

Assim, a identificação de uma conexão da camada “N”, ou N-CI (Connection Identifier), se refere a um específico N-CE de um determinado N-SAP.

A figura A.1 ilustra a relação entre estes conceitos.

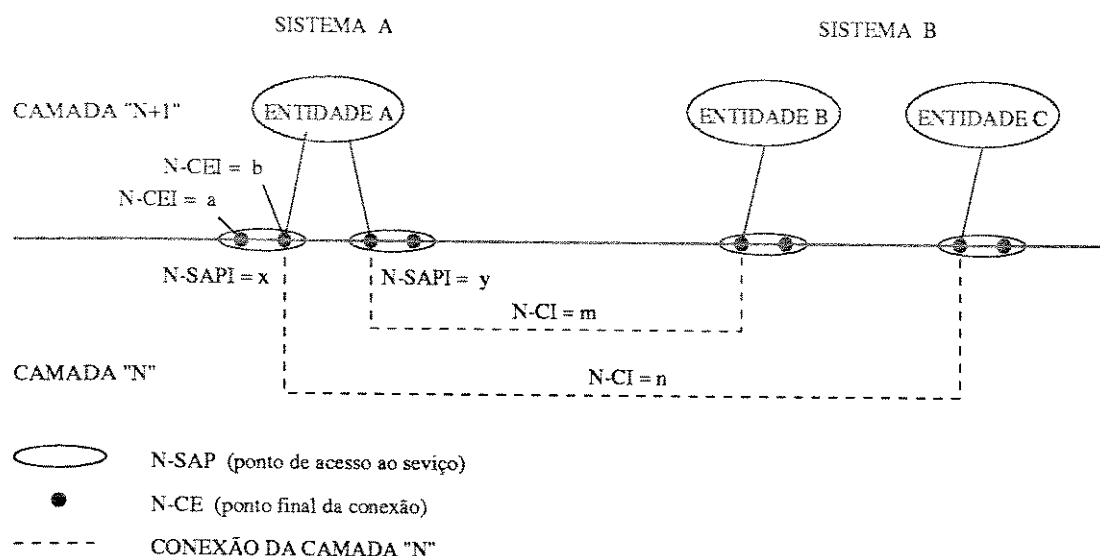


Figura A.1: Relação entre entidades, conexão, SAP e CE

No contexto deste trabalho, por exemplo, as opções de conexões AAL consideradas, podem ser agrupadas em diferentes AAL-SAP que representam classes de qualidade de serviço diferentes. Além disto, como a implementação realizada para a camada AAL não diferencia o serviço prestado por conexão, considera-se que esta diferenciação é realizada pelas camadas ATM (ex. conexões com prioridades diferentes) e FÍSICA (ex. enlaces com taxa de erro diferentes).

Desta forma, as conexões AAL são associadas diretamente às conexões ATM, conforme ilustrado na figura A.2.

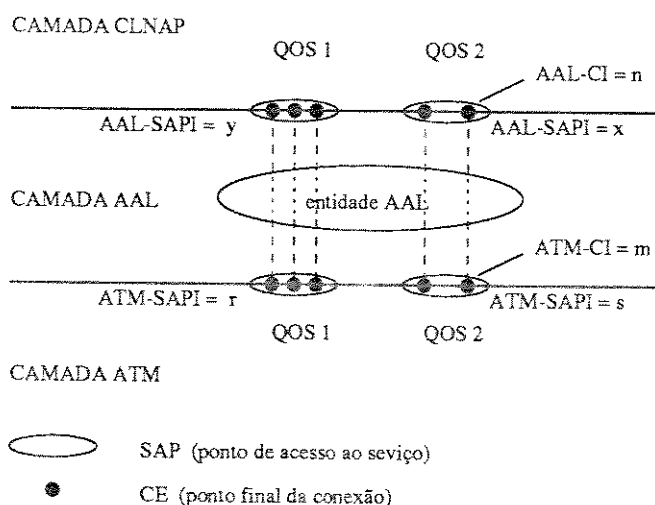


Figura A.2: Associação entre conexões AAL e ATM

Salienta-se que o identificador de uma conexão ATM (ATM-CI) representa logicamente os identificadores VPI e VCI, que definem a conexão ATM virtual, e a identificação da conexão física (FÍSICA-CI), pela qual a célula será encaminhada.

A.2 Primitivas de serviço

O provisionamento do serviço de uma camada está baseado em um conjunto de primitivas de serviços, que representam de uma maneira abstrata a troca de informações entre o usuário e o provedor do serviço da camada.

Existem quatro tipos de primitivas:

- “request”: Gerada pelo usuário para requisição de um serviço ao provedor.
- “indication”: Gerada pelo provedor, ao usuário do serviço, para invocar um procedimento demandado localmente ou remotamente através da primitiva “request”.
- “response”: Gerada pelo usuário para indicar, ao prestador do serviço, o término do procedimento invocado pela primitiva “indication”.
- “confirm”: Gerada pelo provedor para indicar, ao usuário do serviço, o término do serviço requisitado pela primitiva “request”.

A figura A.3 ilustra a relação entre as primitivas de serviço.

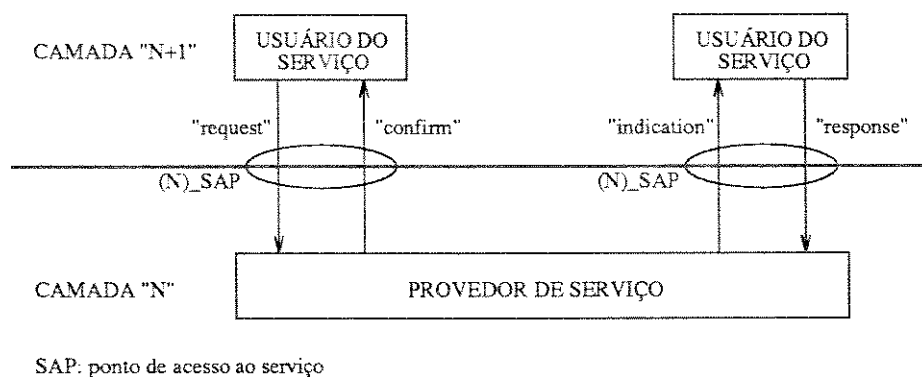


Figura A.3: Primitivas de serviço

Estas primitivas, quando referenciadas neste trabalho, são apresentadas no formato “X_Y_z (lp)”⁴², onde “X” especifica a camada (ex. ATM, AAL, CLNAP), “Y” especifica o serviço (ex. DATA, CONNECT, DISCONNECT), “z” define o tipo de primitiva e “lp” especifica a lista dos parâmetros transmitidos.

A primitiva ATM_DATA_request (ATM_SDU), por exemplo, indica que uma entidade da camada AAL, superior à ATM, está requisitando à camada ATM, a transmissão de uma unidade de dado designada por ATM_SDU.

42. Este formato está baseado no padrão definido pelo CCITT [44].

A.3 Protocolo

A interação entre entidades pares locais e remotas, envolvida na provisão do serviço de uma camada, é denominada por protocolo da camada. A especificação do protocolo descreve a operação de cada entidade da camada em resposta a comandos do usuário do serviço, à ações iniciadas internamente a camada ou a mensagens das entidades pares remotas.

A.4 Unidades de dado

A comunicação entre entidades é realizada através de unidades de dado. As unidades de dado trocadas, entre entidades pares locais e remotas, no processamento do protocolo da camada “N” são denominadas por (N)_PDU (Protocol Data Unit). Estas são constituídas por informações de controle do protocolo, designadas por (N)_PCI (Protocol Control Information)⁴³ e por informações do usuário.

As unidades de dado trocadas entre entidades das camadas “N” e “N+1” são denominadas por (N)_SDU (Service Data Unit). Estas unidades são preservadas durante o transporte através de uma conexão da camada “N”.

A figura A.4 ilustra a relação entre as unidades de dado definidas.

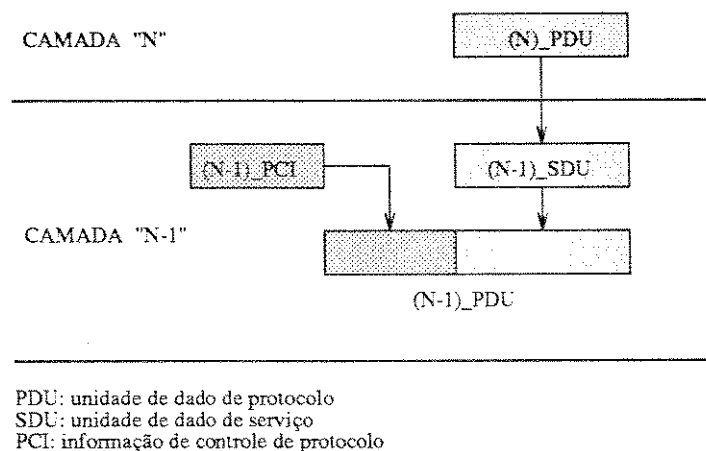


Figura A.4: Relação entre as unidades de dado

43. As informações (N)_PCI podem estar representadas como cabeçalho ou "trailer" da unidade (N)_PDU.

Apêndice B

Características básicas da linguagem SDL

O objetivo deste apêndice é apresentar as características básicas da linguagem SDL, versão 1988 [9], [45] e [46].

B.1 Introdução

SDL (Specification and Description Language) é uma linguagem de descrição formal desenvolvida e padronizada pelo CCITT para especificação e descrição do comportamento de sistemas de telecomunicações. De forma geral, a linguagem SDL se aplica a todos os sistemas cujo comportamento pode ser modelado através de interações entre máquinas de estados finitos estendidas [47].

A linguagem SDL descreve o sistema, de forma gráfica ou textual, em diferentes níveis hierárquicos de abstração, desde a estrutura geral, como por exemplo, a interação entre blocos funcionais do sistema, até em níveis de detalhamento próximos da implementação física do sistema.

O desenvolvimento da linguagem SDL teve início em 1972, produzindo a primeira especificação da linguagem em 1976 (recomendações Z.101 a Z.106), seguida por novas versões em 1980 (recomendações Z.101 a Z.104), 1984 (recomendações Z.100 a Z.104), 1988 e 1992 (recomendação Z.100).

Nas versões de 1984 e 1988 a linguagem foi consideravelmente expandida, viabilizando o desenvolvimento e comercialização de ferramentas de suporte à linguagem SDL. A maioria destas ferramentas [48] são suportadas por computadores pessoais (PCs) e estações de trabalho, e possuem facilidades de edição gráfica, análise sintática e semântica, simulação, validação, tradução para linguagens de alto nível e geração de código objeto. O Apêndice C apresenta uma descrição da ferramenta SDT-2 (SDL Design Tool) [10] utilizada no desenvolvimento deste trabalho.

A versão de 1992 complementa e adiciona novas facilidades [49] ao SDL-88, principalmente visando uma estruturação orientada a objeto e uma melhor descrição dos dados e do comportamento do sistema.

B.2 Conceitos básicos

Este item apresenta conceitos básicos da linguagem SDL-88 (versão 1988), enfatizando a abordagem da descrição gráfica de um sistema.

B.2.1 Estruturação e partição do sistema

O sistema, objeto selecionado para estudo, pode ser descrito em SDL através de elementos ou componentes que definem diferentes níveis hierárquicos de abstração e detalhamento. Os elementos são os seguintes:

- Bloco
- Processo
- Serviço
- Procedimento e macro

Os blocos são os elementos de maior nível de abstração e representam a primeira etapa do particionamento do sistema, descrevendo suas macros funções. O bloco pode ser subdividido novamente em blocos (sub-blocos) se for necessário níveis adicionais de detalhamento.

A próxima etapa de particionamento é a subdivisão dos blocos em processos que são os elementos executores das subfunções do sistema. Esta subdivisão deve ser feita, sempre que possível, contemplando aspectos de desempenho, visto que os processos são elementos dinâmicos que podem ser executados paralelamente. Assim, por exemplo, se por um lado um número maior de processos pode representar uma melhor descrição do bloco, por outro lado, pode implicar em um número excessivo de sinais trocados entre processos e um pior desempenho.

Os processos, por sua vez, podem ser subdivididos em serviços que são subprocessos estáticos que executam de forma seqüencial as funções do processo.

Os procedimentos e macros, elementos que representam o menor nível de abstração do sistema, são utilizados para detalhar partes ou funções dos processos e serviços.

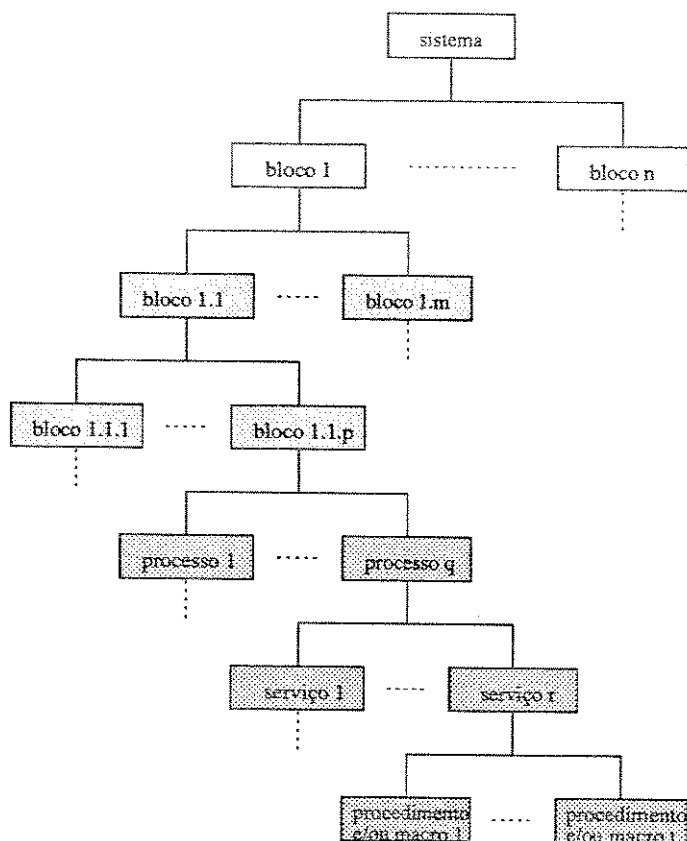


Figura B.1: Hierarquia de descrição do sistema

B.2.1.1 Subestruturas

Subestruturas são elementos do tipo bloco que podem ser adicionados ao sistema com os objetivos de complementação ou de detalhamento da descrição do sistema.

Dois tipos de subestruturas podem ser utilizadas:

- Subestrutura de canal que detalha o comportamento ou funcionalidade da interface entre dois blocos.
- Subestrutura de bloco que detalha ou apresenta outra opção para a descrição do bloco em que está contido.

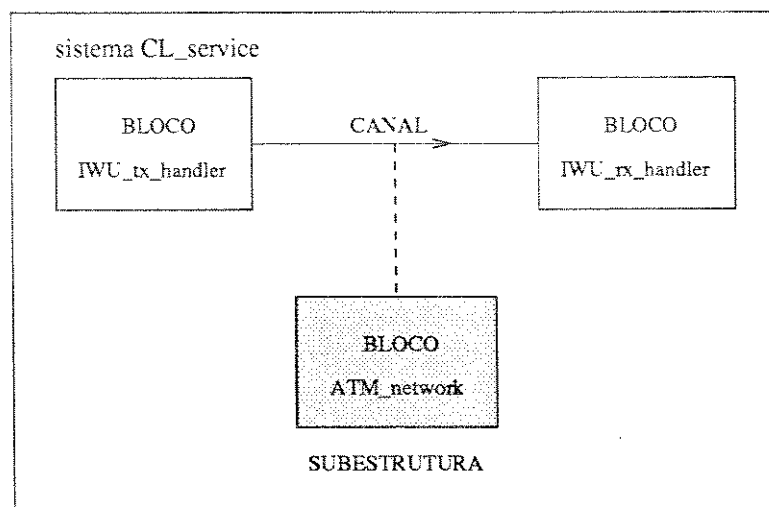


Figura B.2: Subestrutura de canal

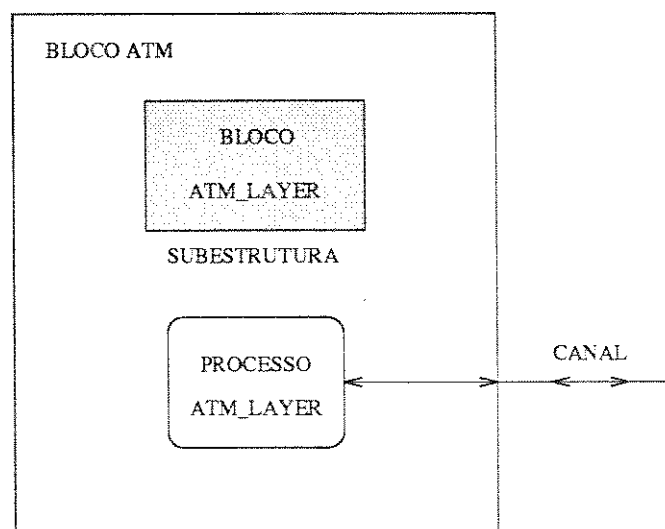


Figura B.3: Subestrutura de bloco

B.2.2 Descrição dos elementos do sistema

Este item apresenta de forma resumida o conteúdo da descrição em SDL, segundo a versão 1988, dos elementos hierárquicos de um sistema.

B.2.2.1 Descrição do sistema

A figura B.4 ilustra a definição do sistema “CLS”.

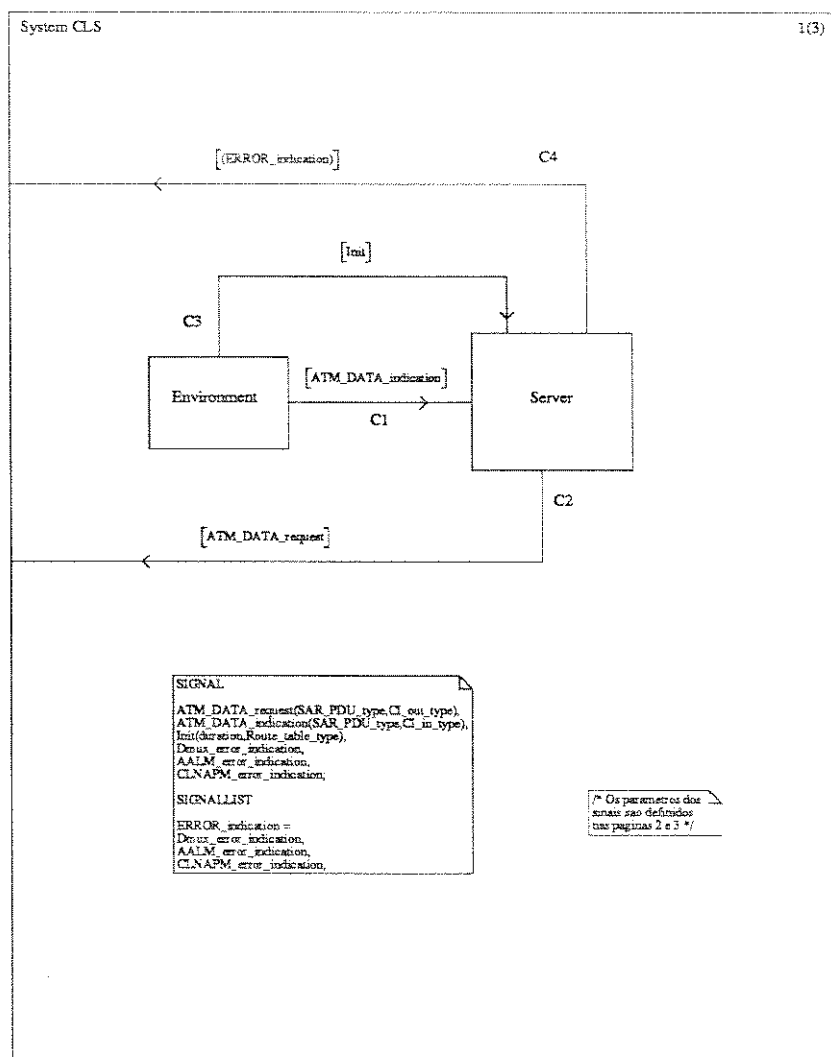


Figura B.4: Sistema “CLS”

A descrição do sistema, que reflete o primeiro nível de detalhamento, inclui os seguintes itens:

- 1- Definição dos blocos e subestruturas que fazem parte do sistema. No exemplo, o sistema “CLS” foi particionado nos blocos “Environment” e “Server” sem a utilização de subestruturas. A descrição da subestrutura segue as mesmas etapas da descrição do bloco.
- 2- Definição das interfaces, ou canais, entre os blocos e destes com o ambiente externo. No exemplo, foram utilizados os canais C1, C2, C3 e C4.

- 3- Definição dos sinais ou listas de sinais associados a cada canal e os tipos dos parâmetros passados com estes sinais. A lista de sinais⁴⁴ é um recurso utilizado para condensar textualmente a descrição de um conjunto de sinais.

No exemplo, o sinal "ATM_DATA_indication" (canal C1) possui dois parâmetros com os tipos "SAR_PDU_type" e "CI_in_type" e a lista de sinais "ERROR_indication" (canal C4) é formada pelos sinais "Dmux_error_indication", "AALM_error_indication" e "CLNAPM_error_indication".

- 4- Definição de dados e tipos que podem ser acessados por todos os blocos do sistema. Estas definições, para este exemplo, são realizadas nas páginas 2 e 3 da descrição do sistema⁴⁵.
- 5- Definição de macros acessíveis por todos os blocos ou subestruturas do sistema. Macros são estruturas, do tipo processo ou procedimento, que são expandidas quando referenciadas.

B.2.2.2 Descrição do bloco

O bloco pode ser particionado novamente em blocos ou em processos. No primeiro caso, a descrição do bloco se procede de forma equivalente à descrição de sistema. No segundo caso, a descrição do bloco é feita conforme ilustrado na figura B.5.

44. A descrição de sinais deve conter, também, os sinais que formam uma lista de sinais.

45. A numeração no canto superior esquerdo indica que o sistema está descrito em três páginas.

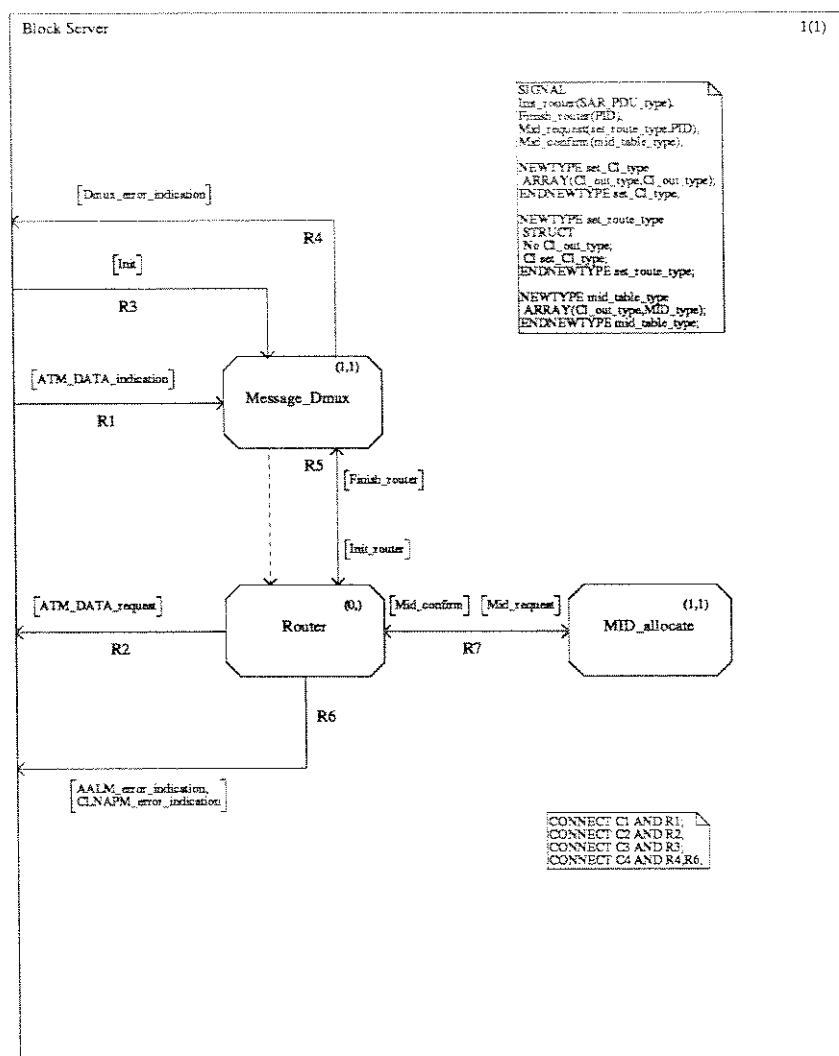


Figura B.5: Bloco “Server”

A definição do bloco contempla os seguintes itens:

- 1- Definição dos processos e subestruturas que fazem parte do bloco e os números inicial e máximo⁴⁶ de instâncias de cada processo. No exemplo, o bloco “Server” é constituído pelos processos “Message_Dmux” e “MID_allocate” de uma única instância e do processo “Router” com um número máximo de instâncias indefinido.
- 2- Indicação dos processos criadores de instâncias. No exemplo, o sinal tracejado indica que o processo “Message_Dmux” cria instâncias do processo “Router”.
- 3- Definição das interfaces, ou rotas, entre os processos e destes com o sistema. No caso do bloco “Server” foram utilizadas as rotas R1 a R7.

46. Estes números são, respectivamente, indicados, entre parênteses, na construção do processo.

- 4- Definição dos sinais ou listas de sinais associados às rotas internas ao bloco e os tipos dos parâmetros passados com estes sinais. Os sinais relativos as interfaces com o sistema, já definidos na descrição do sistema, são aqui apenas referenciados. No exemplo, foram definidos os sinais "Init_router", "Finish_router", "Mid_request" e "Mid_confirm".
- 5- Conexão das interfaces (rotas) do bloco com as respectivas interfaces (canais) definidas na descrição do sistema. No caso do bloco "Server", os canais C1, C2 e C3 estão, respectivamente, conectados às rotas R1, R2 e R3, e o canal C4 às rotas R4 e R5.
- 6- Definição de dados e tipos que podem ser acessados por todos os processos do bloco.
- 7- Definição de macros acessíveis por todos os processos e subestruturas do bloco.

B.2.2.3 Descrição do processo

O processo, que define o comportamento dinâmico do sistema, é descrito conforme ilustrado na figura B.6.

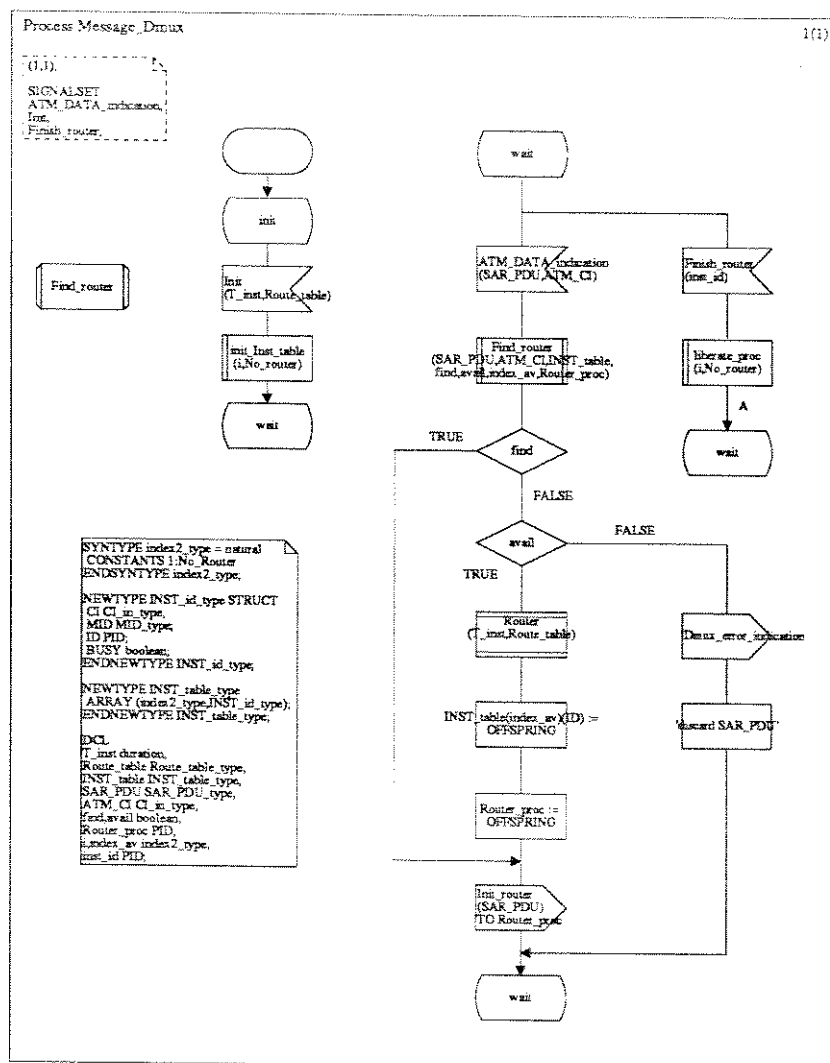


Figura B.6: Processo “Message_Dmux”

A descrição do processo consta dos seguintes itens:

- 1- Indicação dos números inicial e máximo de instâncias do processo, definidos na descrição do bloco. Para o processo “Message_Dmux”, estes números assumem o valor um.
- 2- Indicação dos parâmetros e tipos associados recebidos do processo criador. Estes parâmetros são definidos na descrição do processo criador. Na descrição do processo “Router”, por exemplo, os parâmetros “T_inst” e “Route_table” devem ser indicados⁴⁷.
- 3- Indicação dos sinais recebidos pelo processo, definidos na descrição do sistema ou do bloco do qual o processo faz parte. No exemplo, são indicados os sinais “ATM_DATA_indication”, “Init” e “Finish_router”.

47. Estes parâmetros são declarados na construção “create” do processo “Message_Dmux”.

- 4- Definição do corpo do processo, ou seja, do seu comportamento através das construções SDL de estado, sinais de entrada e saída, tarefa, condição, procedimento, etc.
- 5- Indicação dos procedimentos referenciados no corpo do processo ou referenciados internamente a estes procedimentos. Esta indicação é feita através da construção de referência de procedimento. No exemplo, o procedimento "Find_router" é referenciado.
- 6- Definição dos sinais trocados entre instâncias do mesmo processo.
- 7- Definição de dados e tipos locais ao processo, variáveis acessíveis a outros processos e variáveis importadas de outro processo. No exemplo, as variáveis locais "find" e "avail" são definidas como sendo do tipo "boolean".
- 8- Definição de macros acessíveis no escopo do processo. No exemplo, são definidas as macros "init_inst_table" e "liberate_proc".
- 9- Definição de temporizadores.

O processo pode, também, ser descrito através de serviços. A figura B.7 ilustra a definição do processo "Message_Dmux", particionado nos serviços "Init", "allocate" e "liberate".

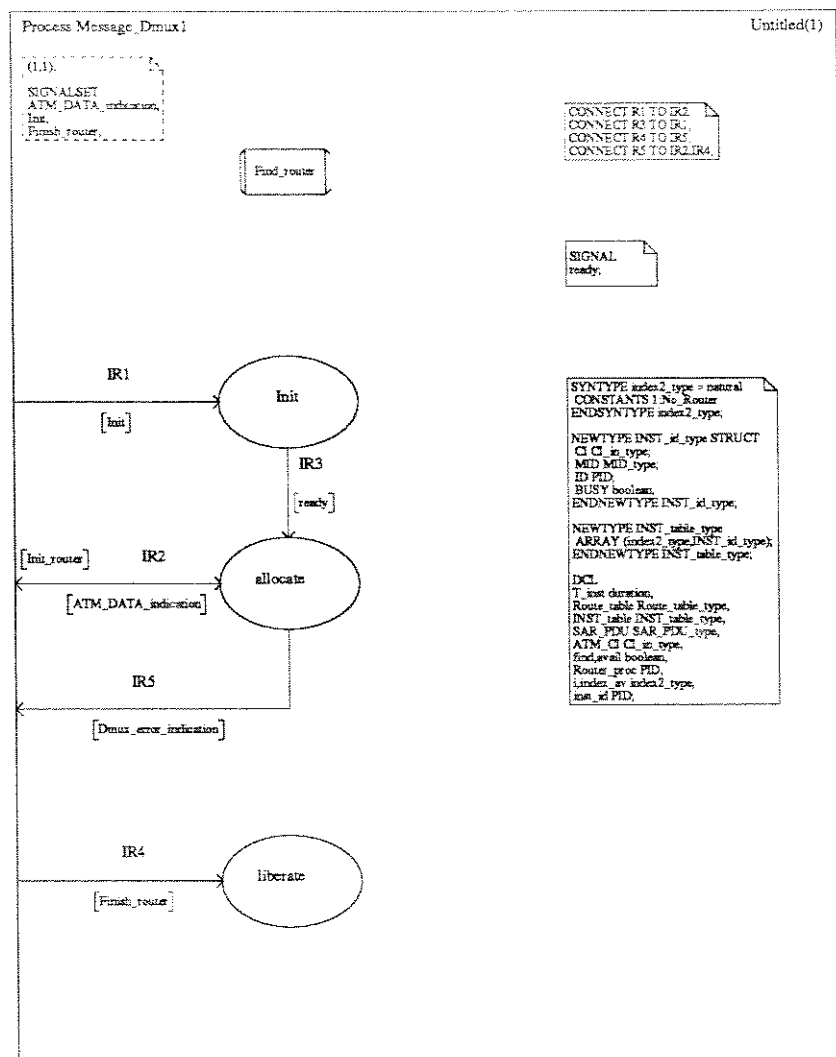


Figura B.7: Processo “Message_Dmux” particionado em serviços

Neste caso, a descrição para o processo apresentada se aplica substituindo-se a definição do corpo do processo pelos seguintes itens:

- Definição dos serviços que fazem parte do processo.
- Definição dos sinais trocados entre os serviços e seus parâmetros. No exemplo, somente o sinal “ready”, sem parâmetros, foi definido.
- Conexões das interfaces internas ao processo, com as respectivas interfaces definidas na descrição do bloco ou do sistema. No processo exemplificado, as rotas R1, R3 e R4 estão, respectivamente, conectados às interfaces IR2, IR1 e IR5, e a rota R5 as interfaces IR2 e IR4.

B.2.2.4 Descrição do serviço

A figura B.8 ilustra a descrição de um serviço.

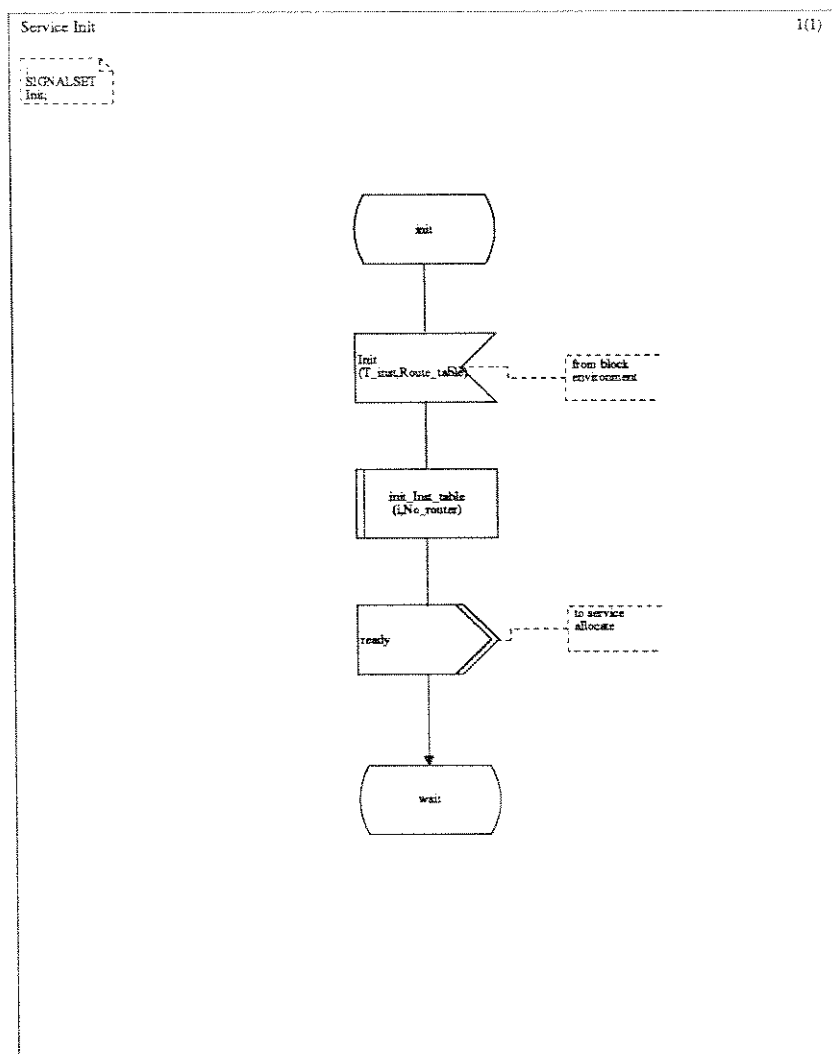


Figura B.8: Descrição do serviço “Init”

A descrição do serviço segue o mesmo procedimento da descrição do processo, aplicada agora ao escopo do serviço, excluindo-se os itens 1, 2 e 6.

Os sinais de entrada e saída, da interface com os demais serviços do mesmo processo, são descritos, respectivamente, pelas construções “priority input” e “priority output”. No exemplo, a construção “priority output” aplica-se ao sinal “ready”. Este sinal é consumido, pelo processo, com prioridade sobre os demais sinais.

B.2.2.5 Descrição de procedimento e macro

A figura B.9 ilustra a definição de um procedimento.

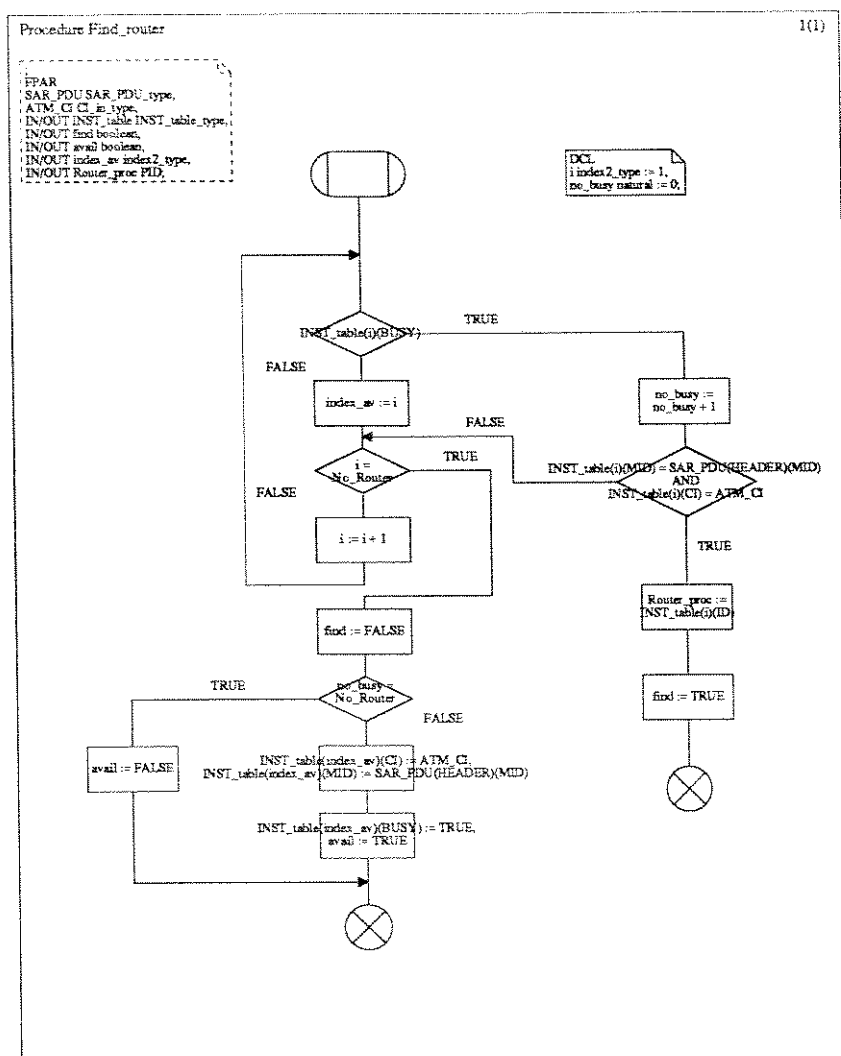


Figura B.9: Definição do procedimento “Find_router”

O procedimento é descrito pelos seguintes itens:

- 1- Indicação dos parâmetros e tipos associados recebidos ou enviados pelo procedimento. No exemplo, são indicados sete parâmetros, dos quais os parâmetros “SAR_PDU” e “ATM_CI” são apenas de entrada e os demais de entrada e saída ao procedimento.
- 2- Definição do corpo do procedimento. Esta é feita de forma semelhante a definição do corpo do processo.

- 3- Indicação dos procedimentos referenciados no corpo do procedimento ou referenciados internamente a estes procedimentos. No exemplo, nenhum procedimento é referenciado.
- 4- Definição de dados e tipos locais ao procedimento. No exemplo, a variável “no_busy” é definida como sendo do tipo “natural” e inicializada com o valor zero.

A figura B.10 ilustra a definição de uma macro.

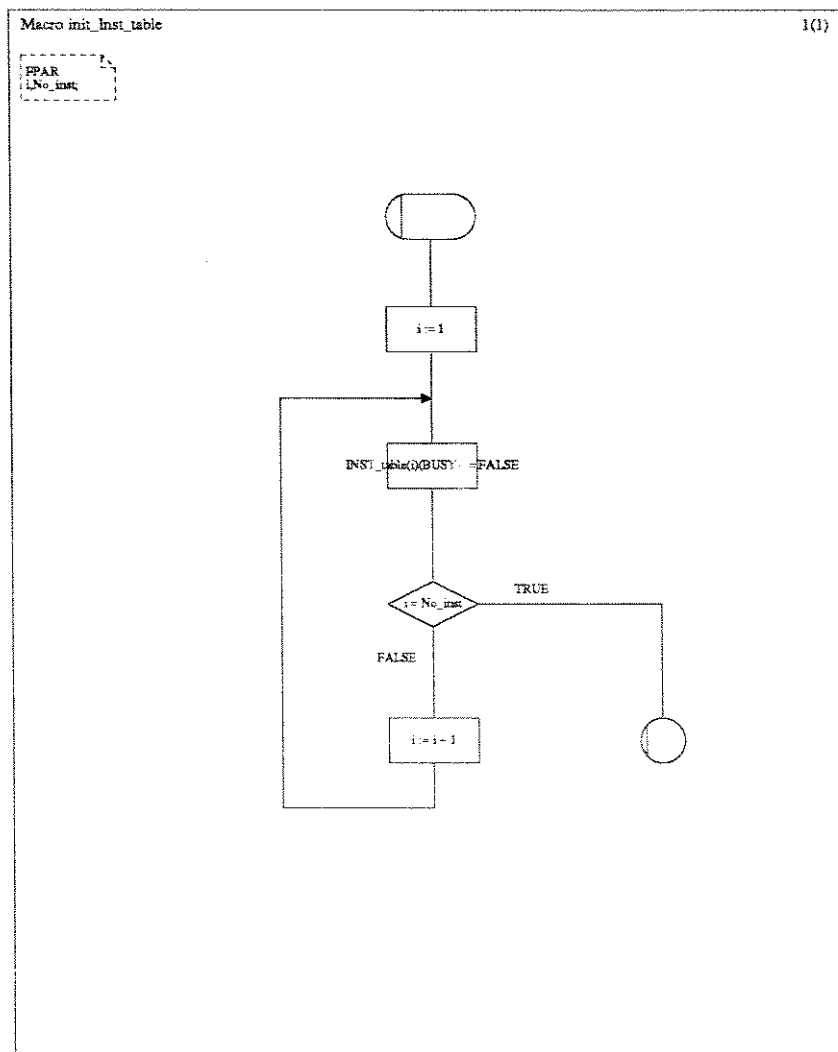


Figura B.10: Definição da macro “init_inst_table”

A macro é descrita através da indicação dos parâmetros recebidos pela macro e a definição de seu corpo. O corpo da macro é definido pelas mesmas construções utilizadas na definição do corpo do processo. No exemplo, a macro recebe os parâmetros “i” e “No_inst”.

B.2.3 Descrição de dados

Os dados na linguagem SDL são descritos através de “tipos abstratos” cuja definição não possui vínculo com a implementação propriamente dita. Esta independência da implementação permite uma maior flexibilidade na descrição dos dados, pois possibilita que novos tipos possam ser definidos, pelo usuário, sem basear-se em outros tipos pré-definidos e sem a preocupação imposta por restrições quanto a representação física de um dado, por exemplo, em 16 ou 32 bits.

A descrição de um tipo em SDL é baseada nos seguintes elementos:

- Valores: representados por números, constantes e variáveis.
- Literais: nomes que representam valores.
- Operadores: definem operações matemáticas aplicadas sobre os valores.
- Expressões: construções envolvendo operadores, valores e literais.

B.2.3.1 Tipos pré-definidos

A linguagem SDL oferece os seguintes tipos pré-definidos:

<i>Boolean:</i>	dois valores denotados pelos literais <i>true</i> e <i>false</i> .
<i>Integer:</i>	números inteiros.
<i>Natural:</i>	números inteiros positivos.
<i>Real:</i>	números inteiros e fracionários.
<i>Character:</i>	caracteres definidos pelo alfabeto internacional (ex: “A”, “B”, “#”, CR, LF).
<i>Charstring:</i>	conjunto de caracteres (ex: “SDL”).
<i>PID:</i>	valores que indentificam as instâncias de um processo.
<i>Time:</i>	números reais que representam instantes de tempo.
<i>Duration:</i>	números reais que representam intervalos de tempo, ou seja, a diferença entre instantes de tempo.

B.2.3.2 Tipos estruturados pré-definidos

Os tipos estruturados pré-definidos são os seguintes:

Struct: estrutura que representa a composição de dois ou mais itens, denominados por campos, que podem ser de tipos diferentes.

Ex.: *struct*

nome *charstring*;
RA *natural*;
curso curso_type;

Array: estrutura que representa um conjunto de itens do mesmo tipo. Esta é definida por dois parâmetros, onde o primeiro caracteriza o tipo que define a localização do item dentro da estrutura, e o segundo que caracteriza o tipo do item.

EX.: *array (natural, PID);*

String: estrutura que representa um conjunto de itens ordenados do mesmo tipo. Esta é definida por dois parâmetros, onde o primeiro caracteriza o tipo do item e o segundo é um literal para designação de conjunto vazio.

EX.: *string (PID, vazio);*

Powerset: estrutura que compreende todos os subconjuntos, que podem ser definidos com os itens de um conjunto do tipo especificado.

EX.: num = {1,2};
 powerset (num) = { { }, {1}, {2}, {1,2} }

B.2.3.3 Operadores pré-definidos

<i>OR</i>	operador lógico “ou”.
<i>XOR</i>	operador lógico “ou exclusivo”.
<i>AND</i>	operador lógico “e”.
<i>NOT</i>	operador lógico “complemento”.
<i>MOD</i> ⁴⁸	operador “módulo”.
<i>REM</i> ⁴⁸	operador “resto”.
<i>IN</i>	questiona a presença de um item dentro de um conjunto.
<i>=</i>	operador de igualdade.
<i>/=</i>	operador de diferença.
<i>></i>	operador maior.
<i>>=</i>	operador maior ou igual.
<i><</i>	operador menor.
<i><=</i>	operador menor ou igual.
<i>+</i>	operador de adição.
<i>-</i>	operador de subtração.
<i>*</i>	operador de multiplicação.
<i>/</i>	operador de divisão.
<i>//</i>	operador de adição de “strings”.

B.2.3.4 Definição de novos tipos

Novos tipos e operadores podem ser definidos a partir de uma combinação dos pré-definidos ou de forma independente destes, conforme a necessidade de descrição do usuário.

48. Os operadores *MOD* e *REM* não estão implementados na versão 1988, constando apenas como palavras reservadas.

A definição de novos tipos são feitas através das declarações NEWTYPE e SYNTYPE e suas declarações auxiliares INHERITS, CONSTANTS, LITERALS, OPERATORS E AXIOMS. Os exemplos abaixo ilustram o contexto onde estas declarações se aplicam:

```
1 – SYNTYPE INT1 = integer
    CONSTANTS 1:100
    ENDSYNTYPE INT1;
```

O tipo INT1 é definido como um tipo sinônimo ao *integer*, porém restrito ao intervalo de 1 a 100.

```
2 – NEWTYPE LISTA
    array (INT1, PID);
    ENDNEWTYPE LISTA;
```

O tipo LISTA é definido como sendo um *array* de 100 itens do tipo *PID*.

```
3 – NEWTYPE INT2
    INHERITS integer
    OPERATORS (“+”, “-”)
    ENDNEWTYPE INT2;
```

O tipo INT2 é definido como tendo as mesmas propriedades do tipo *integer*, porém restrito ao uso dos operadores “+” e “-”.

```
4 – NEWTYPE boolean1
    LITERALS
        true, false;
    OPERATORS
        “NOT”: boolean → boolean;
        “AND”: boolean, boolean → boolean;
    AXIOMS
        NOT true == false;
        NOT false == true;
        true AND false == false;
        false AND false == false;
        false AND true == false;
        true AND true == true;
        true AND false == false;
    ENDNEWTYPE boolean1;
```

O tipo *boolean1* é definido pelos literais *true* e *false*, para os quais são especificados os operadores *NOT* e *AND*, que operam segundo o conjunto de regras (axiomas) especificados.

B.2.4 Comunicação entre processos

A comunicação entre processos é realizada através da troca de sinais e expressa pelas construções “input” e “output” e por uma fila tipo FIFO (First In First Out) de recepção, implicitamente associada a cada processo.

A construção “output” cria um sinal que é colocado no topo da fila do processo destino, se este sinal constar da lista de sinais de entrada. Nesta construção são definidos os

parâmetros transportados pelo sinal e a identificação do processo destino, que pode ser feita das seguintes formas:

- Variável do tipo *PID* definida no processo.
- Expressões pré-definidas do tipo *PID*:
 - self*: identificação do processo corrente, ou seja, que está enviando o sinal.
 - parent*: identificação do processo que criou o processo corrente.
 - offspring*: identificação do último processo criado pelo processo corrente.
 - sender*: identificação do processo que enviou o último sinal consumido.
- Implicitamente, sem especificação, quando existe um único processo destino, definido pela própria estrutura do sistema.

A construção “input” especifica o sinal esperado em um determinado estado do processo. Se o primeiro sinal da fila de recepção coincide com o especificado este é consumido e o processo sai do estado corrente. Caso contrário, o sinal é descartado.

A figura B.11 ilustra a dinâmica da troca de sinais entre processos.

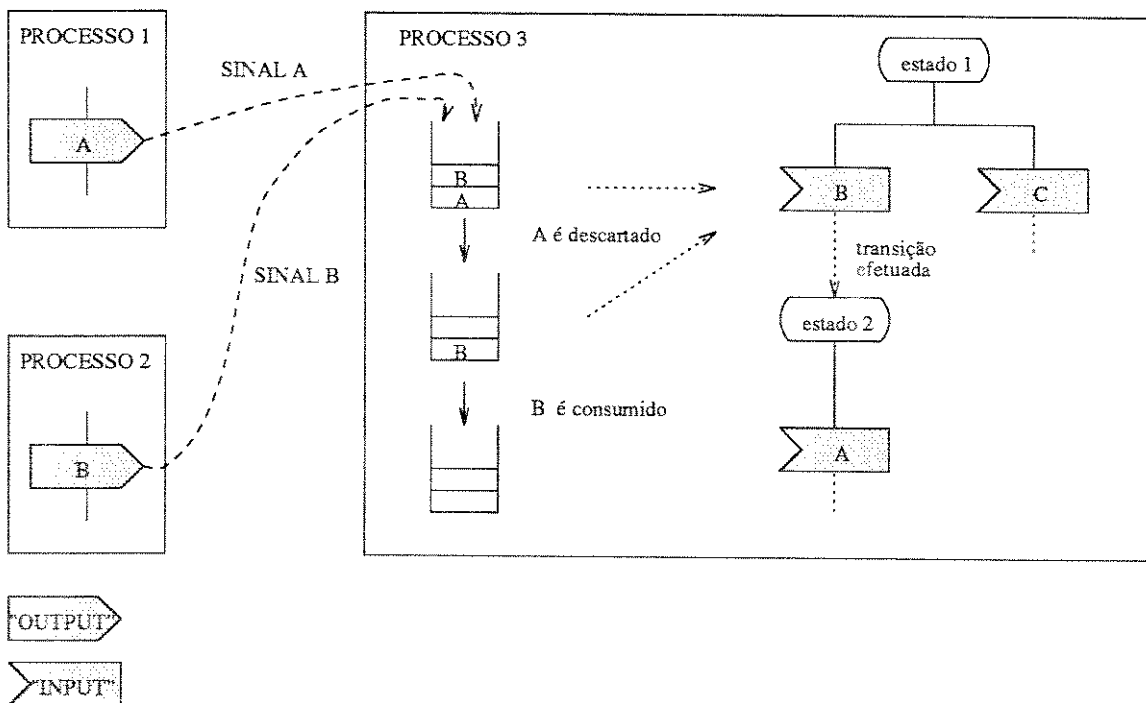


Figura B.11: Dinâmica da comunicação entre processos

Na figura B.11 o sinal “A” é descartado pois o processo 3, quando no estado 1, só prevê a chegada dos sinais “B” e “C”⁴⁹.

Uma restrição da linguagem SDL na dinâmica da comunicação entre processos é que não existe uma maneira formal de priorizar sinais e processos. Assim, para efeitos de descrição estas priorizações podem ser definidas informalmente através das construções utilizadas para comentários.

B.2.5 Descrição de tempo

A descrição de eventos temporais em SDL pode ser feita através do uso de temporizadores e da manipulação de variáveis dos tipos *time* e *duration*. Esta descrição é feita com auxílio da expressão *NOW* que representa o instante de tempo corrente, acessado do relógio de referência do sistema.

A figura B.12 ilustra a manipulação dos tipos *time* e *duration*.

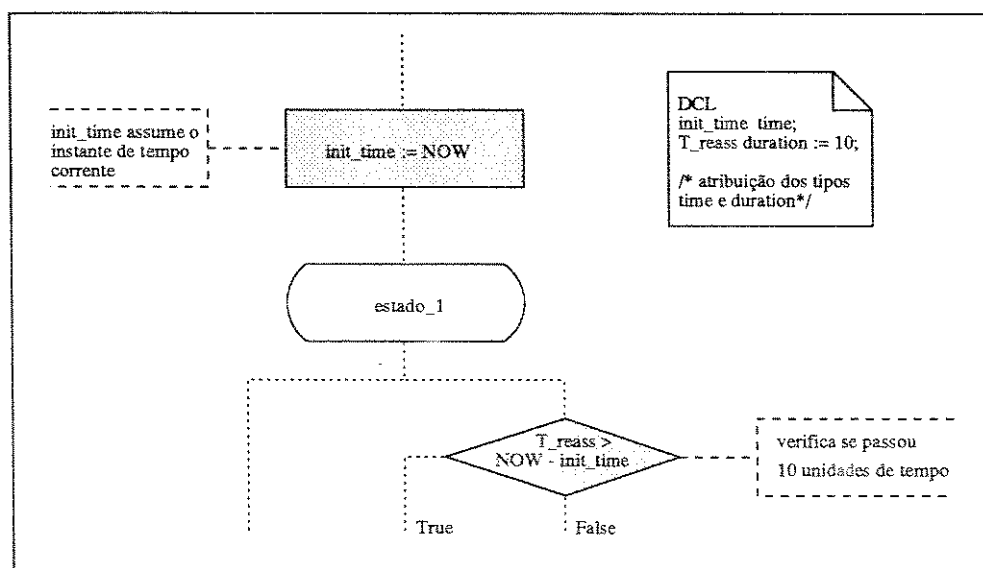


Figura B.12: Manipulação dos tipos *time* e *duration*

O temporizador é um meta-processo que atua dentro do escopo do processo no qual foi definido e que é executado paralelamente a este. A atuação do temporizador é restrita a geração de um sinal, após um intervalo de tempo pré-definido, que é introduzido na fila de recepção do processo.

O temporizador é controlado por duas construções:

SET: ativa o temporizador para gerar um sinal após o intervalo de tempo especificado.

RESET: desativa o temporizador e retira o sinal da fila de recepção se presente.

49. Este descarte pode ser evitado com o uso da construção “save”.

A figura B.13 ilustra o uso destas construções.

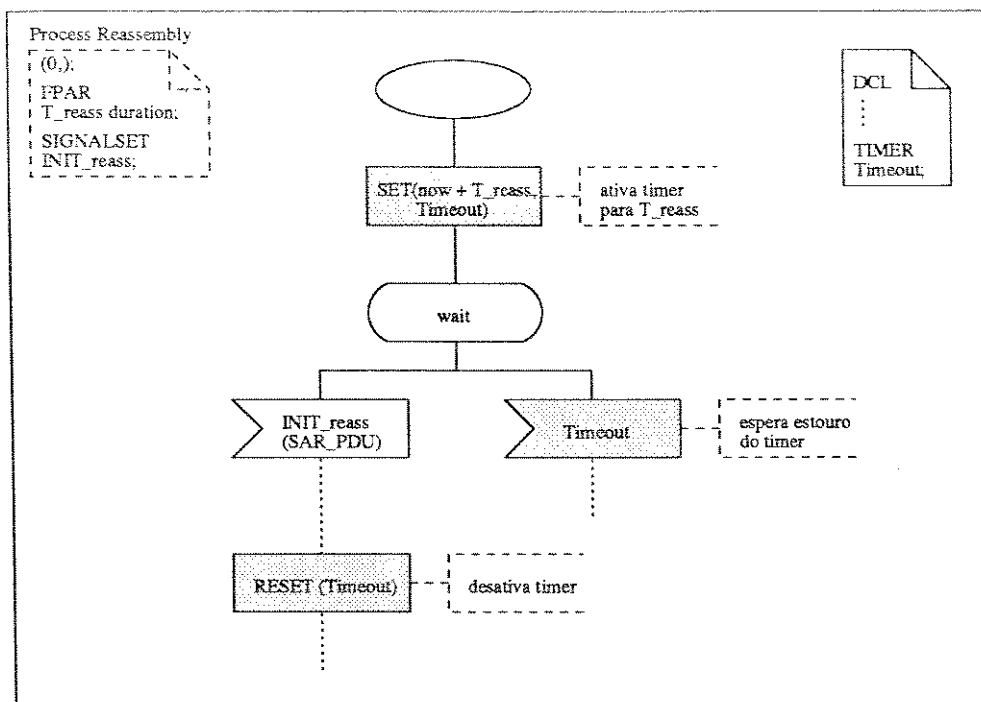


Figura B.13: Definição e controle do temporizador

B.2.6 Construções da linguagem SDL

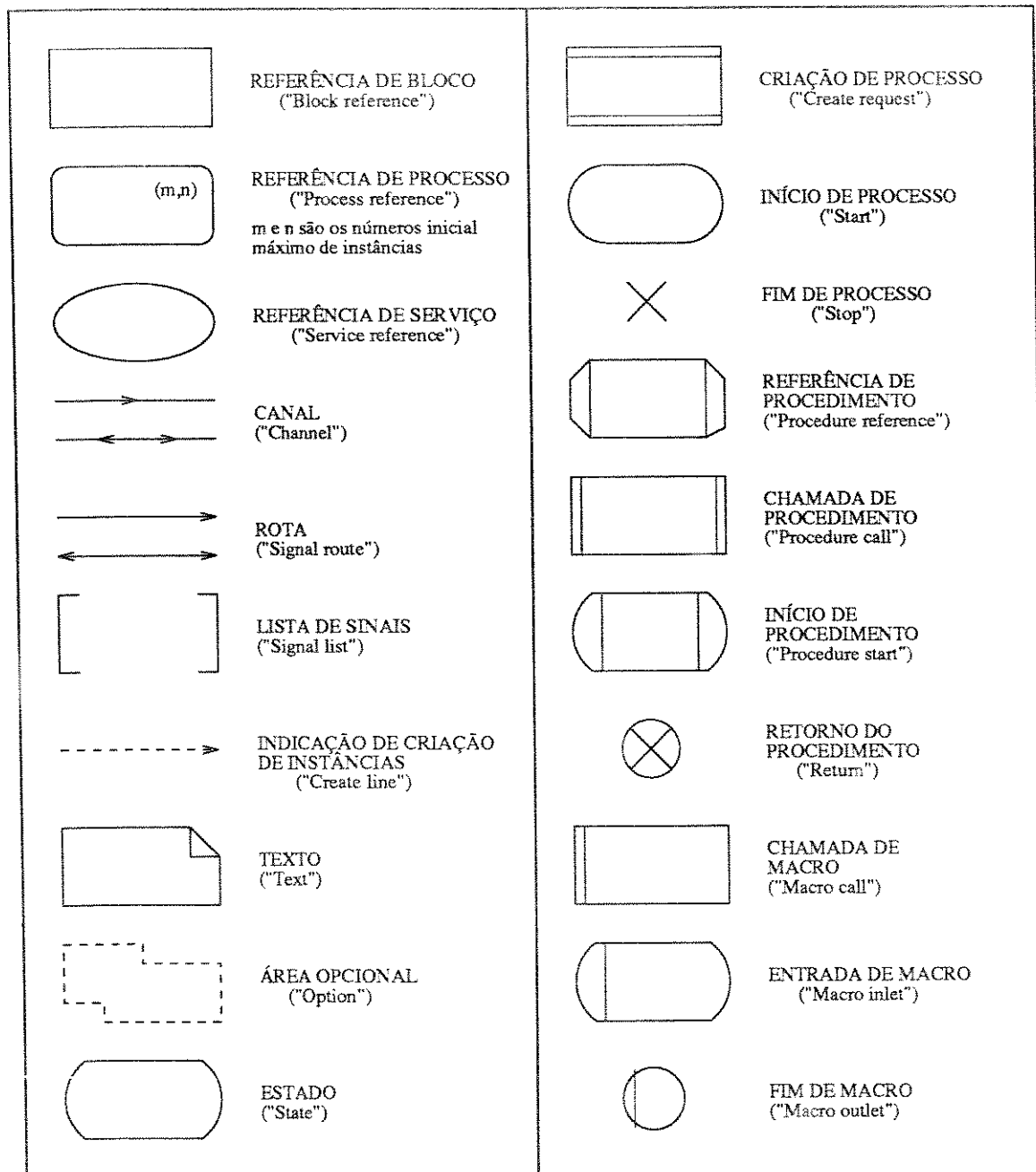


Figura B.14: Construções da linguagem SDL

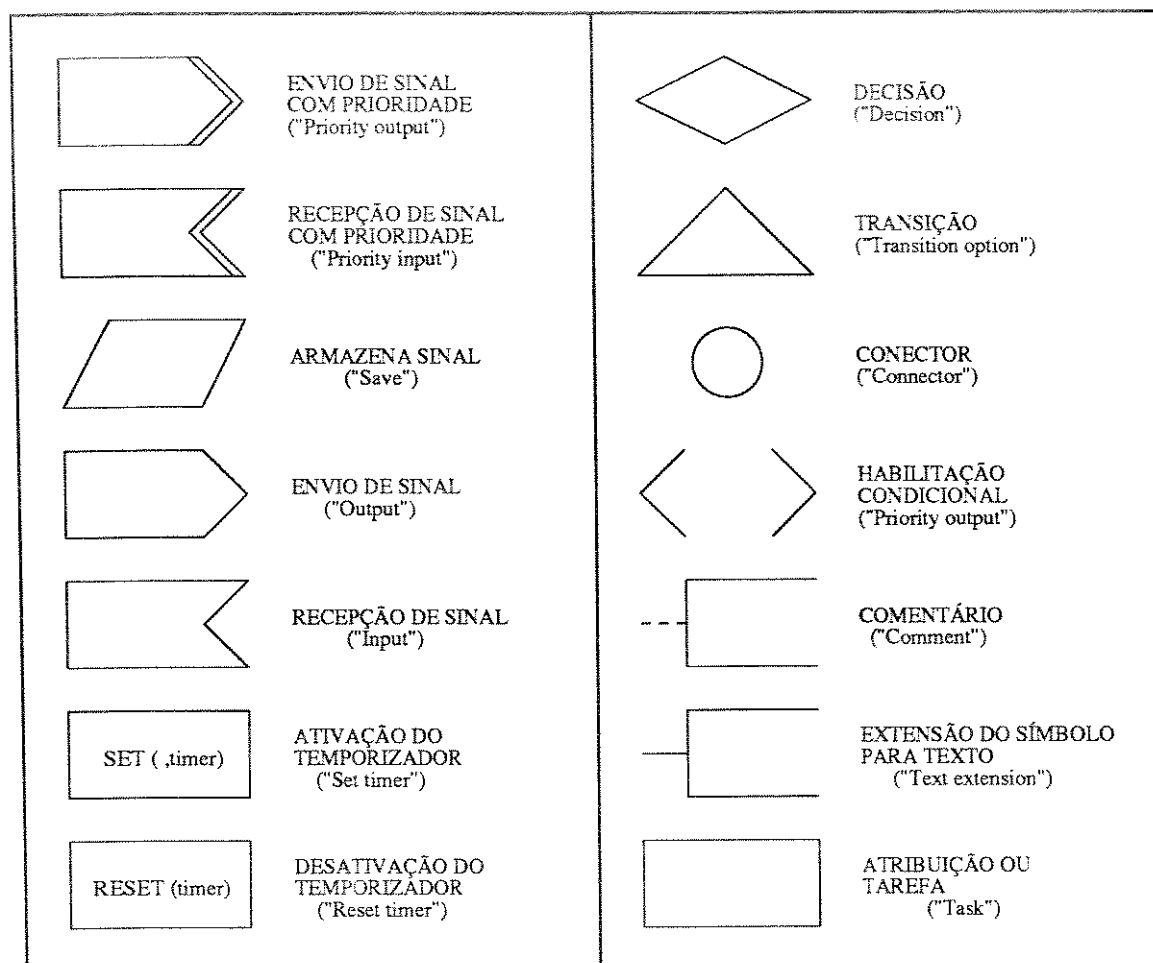


Figura B.15: Construções da linguagem SDL (continuação)

B.3 Restrições da linguagem SDL

As restrições da linguagem SDL, encontradas durante o processo de especificação, e suas implicações são apresentadas a seguir:

- 1- Não previsão da definição formal de prioridade para a execução dos processos e para o tratamento dos sinais de entrada.

Na especificação do bloco "IWU_TX_handler" do sistema IWU, por exemplo, o processo "IP_tx_Dmux" deve priorizar o tratamento do sinal "Finish_ip_tx", que libera a instância do processo "IP_tx", para evitar a perda de sinais "LLC_DATA_indication" por falta de instâncias deste processo.

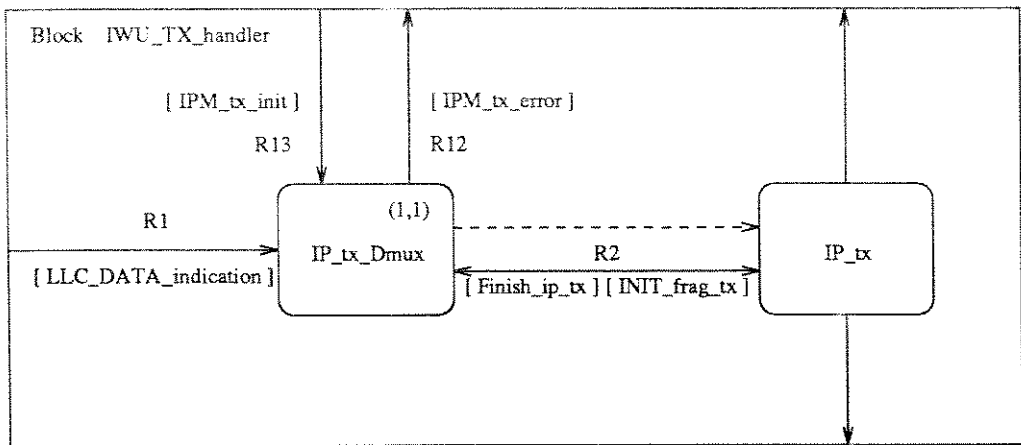


Figura B.16: Interfaces do processo “IP_tx_Dmux”

A ferramenta SDT-2 provê duas maneiras de resolver este problema⁵⁰:

- Durante o processo de simulação do sistema, através de um comando do simulador que atua na fila de sinais dos processos.
- Durante a especificação do sistema, através da primitiva “PRIO”, disponível na ferramenta SDT-2⁵¹ e declarada na linguagem SDL como comentário, conforme ilustrado na figura B.17.

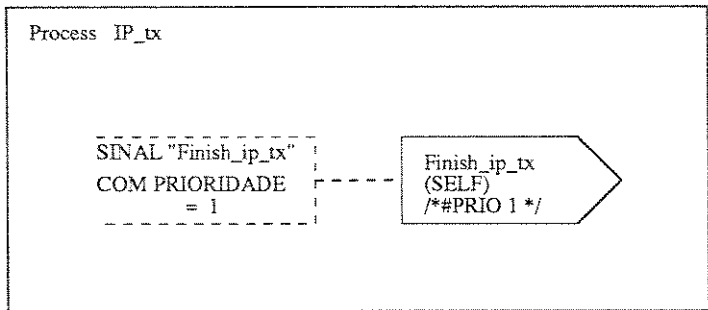


Figura B.17: Definição de prioridade de um sinal

- 2- Não previsão da referência de procedimentos na especificação de sistema e de bloco⁵². Isto implica que procedimentos comuns a blocos ou a processos devem ser descritos repetidamente em cada processo ou definidos como macros.

50. Estas soluções também se aplicam na definição de prioridade para processos.
 51. Esta solução não foi utilizada em virtude da não disponibilidade da “master library” na configuração atual da ferramenta.
 52. Esta restrição foi eliminada na versão de 1992.

No caso do sistema “IWU”, por exemplo, os procedimentos comuns aos blocos “IWU_TX_handler” e “IWU_RX_handler” foram definidos como macros.

- 3- Não implementação dos operadores “MOD” e “REM”⁵². Nas especificações estes operadores foram substituídos por suas definições matemáticas, conforme ilustrado na figura B.18.

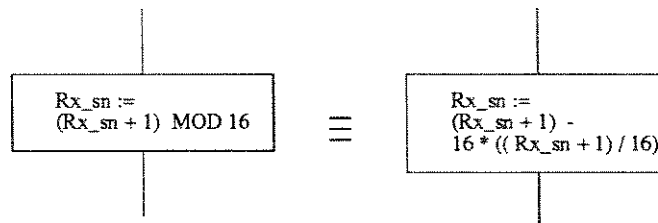


Figura B.18: Exemplo de definição do operador “MOD”

Apêndice C

Ferramenta SDT

A ferramenta SDT⁵³ (SDL Design Tool) [10] é um conjunto integrado de aplicativos software, baseados nas linguagens SDL [9] e MSC (Message Sequence Chart) [50], utilizados para gerenciar e automatizar o desenvolvimento de sistemas. Estes aplicativos apoiam as diferentes etapas do ciclo de desenvolvimento de sistemas, que são:

- a- Definição e especificação
- b- Verificação
- c- Validação
- d- Implementação
- e- Manutenção

C.1 Resumo dos principais aplicativos

A figura C.1 ilustra os principais aplicativos, que compoem a ferramenta SDT, dentro do ciclo de desenvolvimento de um sistema.

53. A ferramenta SDT foi adquirida pelo Departamento de Telemática da FEE – UNICAMP com o apoio da FAPESP através do projeto temático “Redes Metropolitanas Multiserviços” – processo: DT/FEE/UNICAMP – FAPESP Nº 91/3660-0.

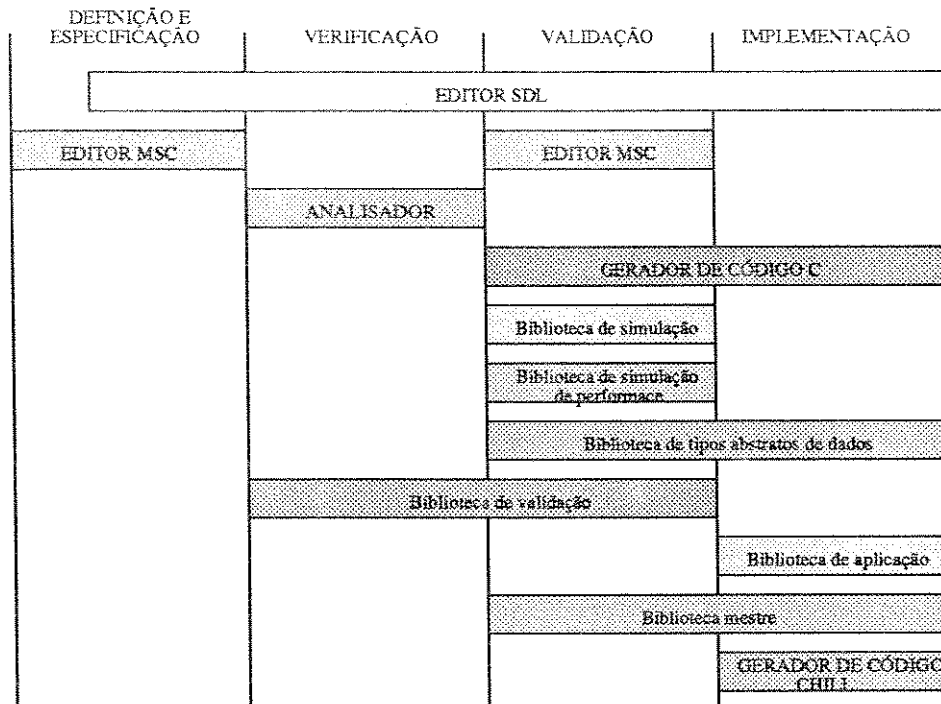


Figura C.1: Aplicativos e bibliotecas da ferramenta SDT

- 1- **Editor gráfico MSC:** Aplicativo complementar a descrição em linguagem SDL, utilizado na etapa de definição e especificação do sistema para documentação e análise de seus aspectos dinâmicos.
Na etapa de validação, diagramas MSC podem ser utilizados pelo simulador para apresentação de “traces” e pelo validador para verificação da consistência do diagrama com a descrição SDL do sistema.
- 2- **Editor gráfico SDL:** Aplicativo utilizado em todas as etapas do desenvolvimento de sistemas, desde a especificação, onde é realizado o particionamento do sistema em blocos, subestruturas, processos, serviços, etc, até a etapa de implementação, onde podem ser realizados o detalhamento e as adaptações da especificação à plataforma de execução.
- 3- **Analisador:** Utilizado na etapa de verificação para exame das especificações realizadas através do editor SDL, em termos do formalismo da linguagem. Este exame compreende:
 - Análise sintática
 - Análise semântica
 - Análise dinâmica

A análise dinâmica indica a possibilidade de erros dinâmicos da especificação (ex. envio de sinal a um processo não criado, uso de variável antes de sua atribuição) e uso

indevido da linguagem SDL (ex. variável declarada mas não utilizada, temporizador definido mas não ativado).

- 4- **Gerador de código C:** Aplicativo que provê a tradução automática da especificação em linguagem SDL para a linguagem C. O código gerado pode ser compilado e ligado a bibliotecas específicas em função da aplicação desejada. As bibliotecas são as seguintes:

Simulação:

Para a geração do simulador, que efetua de maneira interativa a verificação do comportamento dinâmico do sistema.

Simulação de performance⁵⁴:

Para a geração de um simulador otimizado para avaliação de performance do sistema.

Validação⁵⁴:

Para a geração do validador, que efetua de maneira automática ou interativa a verificação da especificação do sistema. Esta verificação compreende a análise dinâmica realizada pelo analisador e a detecção de erros mais complexos, tais como, “dead locks” e “signal race”.

Tipos abstratos de dados:

Biblioteca com tipos auxiliares à descrição do sistema (ex. lista de dados, operadores de propósito geral e dados para geração de números aleatórios).

Aplicação⁵⁴:

Biblioteca de tempo real, usada para adaptação da especificação do sistema à plataforma de desenvolvimento utilizada.

Aplicação mestre⁵⁴:

Biblioteca de tempo real, em código fonte, usada para adaptação da especificação do sistema à plataforma proprietária do usuário.

- 5- **Gerador de código CHILL⁵⁴:** Aplicativo que provê a tradução automática da especificação em linguagem SDL para a linguagem CHILL [51], que o usuário pode compilar, testar e executar através do ambiente CHIPSY (CHILL Integrated Programming System) [52].

C.2 Procedimento utilizado

A figura C.2 apresenta o procedimento adotado, com base nos aplicativos disponíveis na ferramenta SDT, para o desenvolvimento deste trabalho.

54. Não consta da configuração da ferramenta disponível.

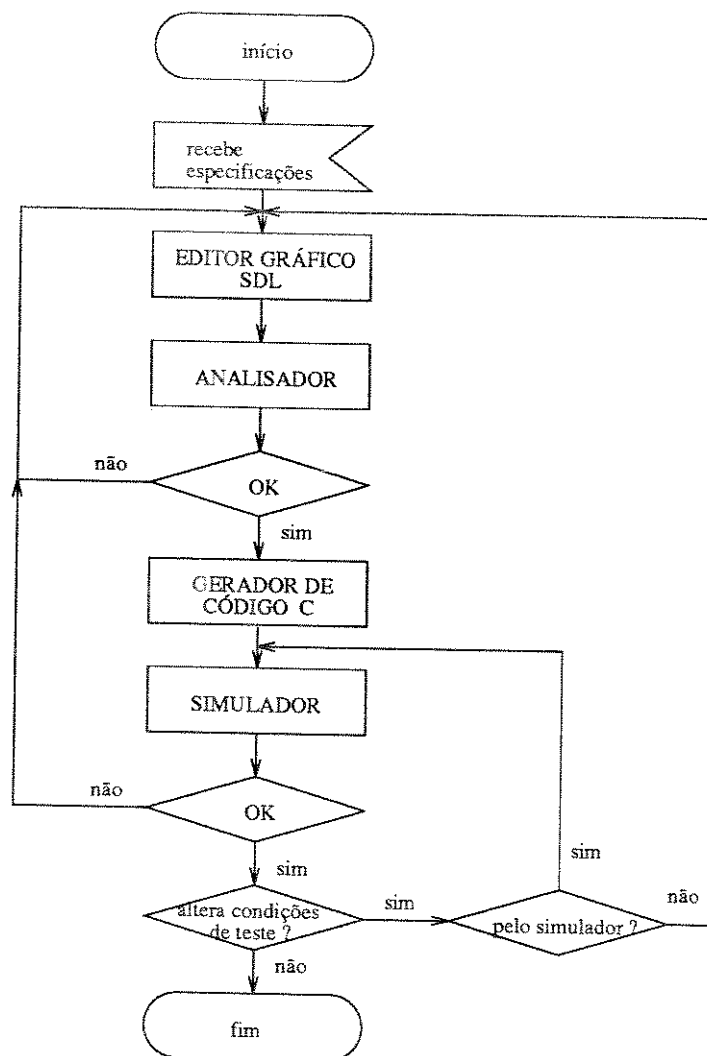


Figura C.2: Procedimento de desenvolvimento

C.3 Compatibilidade com SDL e MSC

A versão utilizada (SDT-2) é compatível com a versão 1988 das linguagens SDL e MSC. As discrepâncias são apresentadas em [10].

C.4 Restrições da ferramenta SDT-2

A principal restrição apresentada pela ferramenta SDT-2 durante o processo de especificação, é que a referência de macros só é permitida na descrição do corpo de um processo.

Na especificação do sistema "IWU", por exemplo, os processos "IP_tx_Dmux" e "IP_rx_Dmux" (pertencentes, respectivamente, aos blocos "IWU_TX_handler" e

“IWU_RX_handler”) implementam as mesmas funções e portanto poderiam ser representados por uma macro, com os parâmetros que diferenciam os dois processos.

Em virtude desta restrição, os processos “IP_tx_Dmux” e “IP_rx_Dmux” foram descritos separadamente e referenciando macros comuns.

C.5 Ambiente utilizado

A plataforma de desenvolvimento utilizada foi a estação de trabalho SUN 4 (SPARC station 10) com sistema operacional SunOS 4.1.x e gerenciador de janelas X-Windows X11R5, instalada no Laboratório de Telemática da Faculdade de Engenharia Elétrica da UNICAMP.

Apêndice D

Glossário

AAL	ATM Adaptation Layer
AAL_CI	AAL Connection Identifier
ATM	Asynchronous Transfer Mode
ATM_CI	ATM Connection Identifier
AL	Alignment
BAS	Buffer Allocation Size
B-ISDN	Broadband-Integrated Services Digital Network
BOM	Beginning of Message
BTAG	Beginning Tag
CASE	Computer Aided Software Engineering
CATV	Cable Television
CE	Connection Endpoint
CEI	Connection Endpoint Identifier
CCITT	International Telegraph and Telephone Consultative Committee (atualmente designado por ITU-TSS)
CI	Congestion Indication
CIB	CRC Indication Bit
CLNAP	Connectionless Network Access Protocol
CLNIP	Connectionless Network Interface Protocol
CLP	Cell Loss Priority
CLS	Connectionless Server
COM	Continuation of Message

Apêndice D : Glossário

COS	Class of Service
CPCS	Common Part Convergence Sublayer
CPI	Common Part Indicator
CRC	Cyclic Redundancy Check
CS	Convergence Sublayer
DA	Destination Address
EFSM	Extended Finit State Machine
EOM	End of Message
ETAG	End Tag
FDDI	Fiber Distributed Data Interface
FIFO	First In First Out
FLA	Flag
FRP	Fast Reservation Protocol
FSM	Finit State Machine
HDTV	High Definition Television
HE	Header Extension
HEL	Header Extension Lenght
HiFi	High Fidelity
HLPI	Higher Layer Protocol Identifier
ID	Identification
IHL	Internet Header Lenght
INFO	Information
IP	Internet Protocol
ISO	International Standard Organization
ITU	International Telecommunication Union
IWU	Interworking Unit
Kbps	Kilo bits per second

LAN	Local Area Network
LI	Lenght Indication
LEN	Lenght
LLC	Logical Link Control
MAC	Medium Access Control
MAN	Metropolitan Area Network
Mbps	Mega bits per second
MID	Multiplexing Identification
MSC	Message Sequence Chart
NMC	Network Management Center
OFF	Offset
OP	Options
OSI	Open System Interconnection
PA SCAD	Pesquisa Aplicada em Sistemas de Comutação de Alto Desempenho
PAD	Padding
PADL	Padding Lenght
PC	Personal Computer
PCI	Protocol Control Information
PDU	Protocol Data Unit
POTS	Plain Old Telephone Service
PRO	Protocol
QOS	Quality of Service
RES	Reserved
SA	Source Address
SAP	Service Access Point
SAR	Segmentation and Reassembly
SDL	Specification and Description Language

SDT	SDL Design Tool
SDU	Service Data Unit
SMDS	Switched Multimegabit Data Service
SN	Sequence Number
SSCS	Service Specific Convergence Sublayer
SSM	Single Segment Message
TCP	Transmission Control Protocol
TOS	Type of Service
TSS	Telecommunication Standardization Sector
TTL	Time to Live
VER	Version
VPI	Virtual Path Identifier
VCI	Virtual Channel Identifier

Referências:

- [1] CCITT Recommendation G.702: Digital Hierarchy Bit Rates, Blue Book, 1988.
- [2] CCITT Recommendation G.704: Synchronous Frame Structures Used at Primary and Secondary Hierarchical Level, Blue Book, 1988.
- [3] CCITT Recommendation G.707: Synchronous Digital Hierarchy Bit Rates, November 1990.
- [4] CCITT Recommendation G.709: Synchronous Multiplexing Structure, November 1990.
- [5] CCITT Recommendation Q.922 Annex A: Core aspects of ISDN Data Link Layer Specification for frame Mode Bear Services, October 1991.
- [6] CCITT Recommendation X.25: Interface Between Data Terminal Equipament (DTE) and Data Circuit-Terminating (DCE) for Terminals Operating in the Packet Mode and Connected to Public Data Networks by Dedicated Circuit, Blue Book, 1988.
- [7] TR-TSV-000772, Generic System Requirements in Support of Switched Multi-Megabit Data Service, Bellcore Technical Reference, Issue 1, May 1991.
- [8] CCITT Recommendation I.150: B-ISDN Asynchronous Transfer Mode Functional Characteristics, November 1990.
- [9] CCITT Recommendation Z.100: Specification and Description Language SDL, Blue Book, 1988.
- [10] SDT 2.3 User's Guide, TeleLOGIC Malmo AB, 1993.
- [11] P. Graef, N. Sakomura, A. Barbieri e C. Klemz. Panorama mundial de redes faixa larga, Nota Técnica, CPqD Telebrás – DDS, Abril de 1993.
- [12] Sistema Trópico RA, Documento interno CPqD Telebrás – DDS, Agosto de 1988.
- [13] CCITT Recommendation I.361: B-ISDN ATM Layer Specification, July 1992.
- [14] CCITT Recommendation I.362: B-ISDN ATM Adaptation Layer (AAL) Functional Description, November 1990.
- [15] CCITT Recommendation I.231: Circuit Mode Bearer Service Categories, Blue Book, 1988.
- [16] CCITT Draft Recommendation Q.93b: B-ISDN User-Network Interface Layer 3 Specification for Basic Call/Bear Control, October 1991.
- [17] Willian Stallings; Handbook of Computer Communications Standards, Vol. 3, Second Edition, 1991.
- [18] CCITT Recommendation I.121: Broadband Aspects of ISDN, November 1990.

- [19] CCITT Recommendation I.321: B-ISDN Protocol Reference Model and its Applications, November 1990.
- [20] CCITT Recommendation X.200: Reference Model of Open Systems Interconnection for CCITT Applications, Blue Book, 1988.
- [21] CCITT Recommendation I.363: B-ISDN Adaptation Layer (AAL) Specification, July 1992.
- [22] CCITT Recommendation X.213: Network Service Definition for Open Systems Interconnection for CCITT Applications, Blue Book, 1988.
- [23] CCITT Draft Recommendation Q.saal0: B-ISDN Signalling ATM Adaptation Layer Overview Description, May 1993.
- [24] CCITT Report COM XVIII-130 Annex 5: Working party XVIII/8 – General B-ISDN Aspects, february 1993.
- [25] CCITT Recommendation I.364: Support of Broadband Connectionless Data Service on B-ISDN, July 1992.
- [26] CCITT Recommendation I.211: B-ISDN Service Aspects, November 1990.
- [27] CCITT Recommendation I.327: B-ISDN Functional Architecture, November 1990.
- [28] L. Mongiovi, M. Farrel and V. Trecordi. A proposal of interconnecting FDDI networks through B-ISDN. INFOCOM Proceedings, Miami, April 1991.
- [29] P. Boyer. A congestion Control for ATM. 7th ITC Workshop Proceedings, Morristow, October 1990.
- [30] P. Crocetti, G. Galassi and M. Gerla. Bandwidth advertising for MAN/ATM connectionless internetting. INFOCOM proceedings, Miami, April 1991.
- [31] ISO 8802-3, IEEE Standard 802.3, Informating Processing Systems – Local Area Network – Part 3: Carrier Sense Multiple Access with Collision Detection (CSMA/CD) Access Method and Physical Layer Specifications, 1990.
- [32] ISO 8802-4, IEEE Standard 802.4, Informating Processing Systems – Local Area Network – Part 4: Token-passing Bus Access Method and Physical Layer Specifications, 1990.
- [33] ISO 8802-5, IEEE Standard 802.5, Informating Processing Systems – Local Area Network – Part 5: Token-ring Access Method and Physical Layer Specifications, 1989.
- [34] ISO 8802-6, IEEE Standard 802.6, Informating Processing Systems – Local Area Network – Part 6: Distributed Queue Dual Bus (DQDB) Subnetwork of a Metropolitan Area Network (MAN), 1991.
- [35] ISO 8802-2, IEEE Standard 802.2, Informating Processing Systems – Local Area Network – Part 2: Logical Link Control, 1989.

- [36] CCITT Recommendation E.164: Numbering Plan for the ISDN Era, Blue Book, 1988.
- [37] CCITT Recommendation I.364 Appendix 1: CRC32 Generation and Checking, July 1992.
- [38] Mark A. Miller, P. E.; Troubleshooting TCP/IP, M&T Books, EUA, 1992.
- [39] ISO IS 9074, Estelle – A formal description technique based on an Extended State Transition Model, ISO/IEC, 1989.
- [40] ISO IS 8807, LOTOS – A formal description technique based on the Temporal Ordering of Observation Behaviour, 1989.
- [41] Pedro Graef Jr; Especificação funcional de um adaptador de protocolo LAN – ATM para o serviço não orientado à conexão na B-ISDN, Nota Técnica 51/94, CPqD Telebrás – DDS, Março de 1994.
- [42] Pedro Graef Jr; Especificação funcional de um servidor para o serviço não orientado à conexão na B-ISDN, Nota técnica 50/94, CPqD Telebrás – DDS, Março de 1994.
- [43] ISO 7498-1984, Information Processing Systems – Open System Interconnection – Basic Reference Model, American National Standards Association, Inc., NY.
- [44] CCITT Recommendation X.210: Layer Service Definition Conventions of Open Systems Interconnection for CCITT Applications, Melbourne, 1988.
- [45] R. Saracco, J. R. Smith. Telecommunications Systems Engineering using SDL, 1989.
- [46] Ferenc Belina, Dieter Hogref and Amardeu Sarma. SDL with Applications from Protocol Specification, Prentice Hall International (UK) Ltda, 1991.
- [47] CCITT Recommendation Z.100 Annex D: SDL User Guidelines, Blue Book, 1988.
- [48] S.T. Chanson, A.A.F. Loureiro, S.T. Vuong; On tools supporting the use of formal description techniques in protocol development; Computer Networks and ISDN Systems, 1993.
- [49] Ove Faergemand and Anders Olsen; New features in SDL-92, TFL Telecommunications Research Laboratory – Denmark, February 1993.
- [50] CCITT Recommendation Z.120: Message Sequence Chart MSC, Blue Book, 1988.
- [51] CCITT Recommendation Z.200: High Level Language CHILL, Blue Book, 1988.
- [52] CHILL Integrated Program System, KVATRO A/S.