

EDUARDO CAMBRUZZI

**AVALIAÇÃO EXPERIMENTAL DA INFRAESTRUTURA COMPUTACIONAL
PARA SISTEMAS INTELIGENTES DE TRANSPORTE**

**FLORIANÓPOLIS
2003**

**UNIVERSIDADE FEDERAL DE SANTA CATARINA
PROGRAMA DE PÓS-GRADUAÇÃO
EM ENGENHARIA ELÉTRICA**

**AVALIAÇÃO EXPERIMENTAL DA INFRAESTRUTURA COMPUTACIONAL
PARA SISTEMAS INTELIGENTES DE TRANSPORTE**

Dissertação submetida à
Universidade Federal de Santa Catarina
Como parte dos requisitos para a
Obtenção de grau de Mestre em Engenharia Elétrica.

EDUARDO CAMBRUZZI

Florianópolis, dezembro 2003.

AVALIAÇÃO EXPERIMENTAL DA INFRAESTRUTURA COMPUTACIONAL PARA SISTEMAS INTELIGENTES DE TRANSPORTE

Eduardo Cambuzzi

‘Esta Dissertação foi julgada adequada para obtenção do Título de Mestre em Engenharia Elétrica, Área de Concentração em *Automação e Sistemas*, e aprovada em sua forma final pelo Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal de Santa Catarina.’

Prof. Werner Kraus Junior, Ph. D.
Orientador

Coordenador da Pós, Prof. Jefferson Luiz Brum Marques Ph. D.
Coordenador do Programa de Pós - Graduação em Engenharia Elétrica

Banca Examinadora :

Prof. Werner Kraus Junior, Ph. D.
Presidente

Prof. Bernardo Gonçalves Riso, Dr.

Prof. Carlos Barros Montez, Dr.

Para Júlia, por sorrir sempre.

AGRADECIMENTOS

Agradeço a todas as pessoas que de alguma forma ofereceram auxílio e suporte no decorrer deste trabalho, principalmente ao prof. Werner Kraus Junior, pela atenção, orientação e paciência dispensados. Agradeço a meus pais, especialmente à minha mãe por seu incentivo e a minha esposa, por suportar a ausência e o constante mau humor. Agradeço também ao amigo Lion, por permanecer e resistir até o final.

Resumo da Dissertação apresentada à UFSC como parte dos requisitos necessários para a obtenção do grau de Mestre em Engenharia Elétrica

AVALIAÇÃO EXPERIMENTAL DA INFRAESTRUTURA COMPUTACIONAL PARA SISTEMAS INTELIGENTES DE TRANSPORTE

Eduardo Cambuzzi

Dezembro/2003

Orientador: Prof. Werner Kraus Junior, Ph. D.

Área de Concentração: Automação e Sistemas

Palavras Chave: Sistemas Inteligentes de Transporte, Protocolos, Padrões Abertos, Interoperabilidade, Intercambialidade.

Número de Páginas: 95

Este trabalho estuda as tecnologias de comunicação entre dispositivos para Sistemas Inteligentes de Transporte – SIT. Visa-se solucionar a falta de interoperabilidade que surge da inexistência de padrões de comunicação para dispositivos SIT. Examina-se em particular os protocolos definidos pelo padrão norte americano *National Transportation Communication for ITS Protocol* – NTCIP para a comunicação entre dispositivos de campo e centrais de controle. Seguindo a especificação do NTCIP, aplica-se o protocolo *Simple Network Management Protocol* - SNMP para a gerência de controladores semafóricos conectados em rede a uma central de controle de tráfego.

Implementa-se o protótipo de um sistema de controle atuado pelo tráfego, consistindo de controladores virtuais e de uma central de controle de tráfego, desenvolvido para testar as funcionalidades do protocolo em um ambiente de microsimulação. Neste ambiente fornecido pelo software SITRA-B+, são recriadas situações de tráfego onde os agentes do sistema (controladores, central) encontram-se distribuídos e podem ser manipulados remotamente.

Através deste estudo, avalia-se a aplicabilidade dos conceitos, padrões e tecnologias descritos pelo NTCIP à realidade nacional, concluindo-se pela necessidade do uso de um protocolo com menor exigência em termos de largura de banda para a comunicação entre os controladores semafóricos e a central de tráfego. Como perspectiva, sugere-se o estudo do protocolo *Simple Transportation Management Protocol* - STMP, descrito na especificação do NTCIP o qual reduz significativamente o tamanho das mensagens.

Abstract of Dissertation presented to UFSC as a partial fulfillment of the requirements for the degree of Master in Electrical Engineering.

EXPERIMENTAL EVALUATION OF COMPUTATIONAL INFRASTRUCTURE TO INTELLIGENT TRANSPORTATION SYSTEMS

Eduardo Cambuzzi

December/2003

Advisor: Werner Kraus Junior, Ph. D.

Area of Concentration: Automation and Systems

Key words: Intelligent Transportation Systems, Protocols, Standards Opened Interoperability, Interchangeability,

Number of Pages: 95

This work studies the technologies for communication among devices for Intelligent Transportation Systems – ITS. It aims at solving the lack of interoperability that arises due to the absence of communication standards for ITS devices. In particular, the protocols defined in the North American standard named National Transportation Communication for ITS Protocol – NTCIP – for communication between field devices and management centers are examined. Following the specification of the NTCIP, the Simple Network Management Protocol – SNMP – protocol is applied to the management of semaphore controllers connected in a network with the management center.

A prototype of a traffic-actuated, real-time traffic control system, consisting of virtual controllers and a traffic management center, is implemented to test the functionalities of the protocol in a microsimulation environment. In this environment, supported by the SITRA-B+ software, traffic situations are recreated where the agents of the system (controller, management center) are distributed and can be manipulated remotely.

Through this study, it is evaluated the applicability of the concepts, standards and technologies of the NTCIP to the Brazilian reality, concluding for the necessity of the use of a protocol with lesser requirements in terms of bandwidth for the communication between semaphore controllers and traffic management centers. As a perspective, the study of the alternative Simple Transportation Management Protocol – STMP –, described in the NTCIP specification, is suggested for significantly reducing the size of the messages.

SUMÁRIO

LISTA DE FIGURAS	xii
LISTA DE TABELAS	xiii
LISTA DE QUADROS	xiv
LISTA DE ABREVIATURAS	xv
1. INTRODUÇÃO.....	1
1.1. Considerações gerais	1
1.2. Objetivos	3
1.2.1. Objetivo Geral	3
1.2.2. Objetivos Específicos	3
1.3. Estrutura do trabalho	3
2 . SISTEMAS INTELIGENTES DE TRANSPORTE.....	5
2.1 Introdução	5
2.2. Estrutura Operacional dos SIT	5
2.2.1. Sistema De Informação Ao Viajante	5
2.2.2. Sistema de gerenciamento do transporte público.....	6
2.2.3. Sistema de Gerenciamento do Transporte De Cargas	6
2.2.4. Sistema de Gerenciamento de Estacionamentos	7
2.2.5. Sistema de Controle de Tráfego.....	7
2.3. Arquiteturas de Referência para SIT.....	9
2.3.1. O Modelo de Arquitetura para Sistemas SIT Norte-Americano	10
2.3.2. O Modelo Europeu de Arquitetura SIT	12
2.4. Considerações Finais	13
3. A PROPOSTA DE ARQUITETURA DE REFERÊNCIA DA <i>NATIONAL ITS ARCHITECTURE</i>	14
3.1. Introdução	14
3.2. Arquitetura Lógica.....	15
3.3. Arquitetura Física	15
3.3.1. Camada Institucional.....	16
3.3.2 Camada de Comunicação.....	16
3.3.3. Camada de Transporte.....	17
3.4. Subsistemas da Arquitetura física.....	18
3.4.1. Subsistema de Centrais	18
3.4.2. Subsistema de Rodovias	20
3.4.3. Subsistema de Veículos	20
3.4.4. Subsistema de Viajantes	21

3.5. Considerações Finais	22
4. A INFRAESTRUTURA DE COMUNICAÇÃO DO NTCIP	23
4.1. Introdução	23
4.2. Níveis Operacionais do NTCIP	25
4.2.1. Nível de Informação	26
4.2.2. Nível de Aplicação	27
4.2.3. Nível de Transporte, Sub-Rede e Físico.....	33
4.3. Aspectos da Comunicação entre Centrais (<i>Center-to-Center</i>) e <i>com seus Dispositivos</i>	33
4.4. Considerações Finais	35
5. ESTUDO DE CASO PARA A IMPLEMENTAÇÃO DO NTCIP	37
5.1. Introdução	37
5.2. Descrição da Aplicação	37
5.3. Arquitetura da Aplicação	39
5.3.1. Nível de Informação	41
5.3.2. Nível de Aplicação	43
5.3.3. Nível de Transporte, Sub-rede e Físico	44
5.4. Arquitetura Interna.....	44
5.4.1. Implementação da Classe MIB.....	44
5.4.2. Implementação da Classe Codificar	46
5.4.3. Implementação da Classe Comunicar	47
5.5. Interface do Usuário	48
5.6. O Ambiente de Simulação	49
5.7. Considerações Finais	51
6. CONCLUSÕES E PERSPECTIVAS FUTURAS	52
7. REFERÊNCIAS BIBLIOGRÁFICAS	54
Anexo 1	57
Anexo 2	75
Anexo 3	79
Anexo 4	82
Anexo 5	84
Anexo 6	86

LISTA DE FIGURAS

Figura 2. 1– Diagrama de Contexto do Sistema de Gerenciamento de Transporte Público e Tráfego Urbano (TAP, 1999).....	12
Figura 3. 1 – Simplificação da NIA.....	14
Figura 3.2 - Descrição em Alto Nível da Arquitetura Lógica (FHWA, 1999c).....	15
Figura 3.3 - Camadas da arquitetura Física da <i>National ITS Architecture</i> (FHWA, 1999b)	16
Figura 3.4 - Visão geral dos serviços de telecomunicação para <i>National ITS Architecture</i> (FHWA, 1999d)	17
Figura 3.5 - Subsistemas e Comunicação da <i>National ITS Architecture</i> (FHWA, 1999d)..	18
Figura 4.1 – O NTCIP e a <i>National ITS Architecture</i> (FHWA, 1999d).....	23
Figura 4.2 – Conjunto de padrões de comunicação do NTCIP (FHWA, 1999d).....	24
Figura 4.3 – Relação entre níveis e camadas dos modelos NTCIP e OSI/ISO.....	26
Figura 4.4 – Objeto de dados NEMA (FHWA, 1999d)	27
Figura 4.5 – Protocolos de Comunicação <i>Center-to-Field</i>	29
Figura 4.6 – Processo para determinar protocolo a ser utilizado (FHWA, 2002b).	32
Figura 4.7 – Exemplo de integração de um SIT usando NTCIP.....	34
Figura 4.8– Migração de um sistema legado /proprietário para um NTCIP puro.....	35
Figura 5.1 – Ilustração dos dispositivos envolvidos na aplicação NTCIP.....	38
Figura 5.2 – Pilha de protocolos NTCIP usados na Aplicação (FHWA, 1999d).....	39
Figura 5.3 – Diagrama de Fluxo de Dados da aplicação	40
Figura 5.4 – Estrutura da árvore MIB para a aplicação.....	42
Figura 5.5 – Figura 5.5 – DFD do processo de Comunicação de um agente SNMP.....	43
Figura 5.6 – Estrutura de uma mensagem SNMP.....	47
Figura 5.7 – Interface gráfico dos processos Gerente e Agente SNMP.....	48
Figura 5.8 – Interface gráfico dos processos Gerente e Agente SNMP.....	49

LISTA DE TABELAS

Tabela 4.1 – Objetos de dados definidos pela NEMA (FHWA, 1997b).....	26
Tabela 4.2 – Comparação dos protocolos STMP,SNMP e SFMP (FHWA, 1999b).....	28
Tabela 4.3 – Definição e Configuração de Objetos Dinâmicos (FHWA, 2002b).....	31

LISTA DE QUADROS

Quadro 5.1 – Código da estrutura de dados de uma MIB.....	45
---	----

LISTA DE ABREVIATURAS

AASHTO - *American Association of State Highway and Transportation Officials*

ASN.1 – *Abstract Syntax Notation - One*

BER – *Basic Encoding Rules*

CORBA – *Common Object Request Broker Architecture*

DATEX – *Data Exchange in ASN.1*

DOD - *Department of Defense*

FHWA – *Federal Highway Administration*

FTP – *File Transportation Communication*

IAB – *Internet Architecture Board*

IETF – *Internet Engineering Task Force*

IP – *Internet Protocol*

ISO – *International Organization for Standardization*

ITE - *Institute of Transportation Engineers*

ITS – *Intelligent Transportation Systems*

MIB – *Management Information Base*

NEMA - *National Electrical Manufactures Association*

NIA – *National ITS Architecture*

NTCIP - *National Transportation Communications for ITS Protocol*

OER - *Octet Encoding Rules*

OID – *Object Identifier*

OSI – *Open System Interconnection*

PDU – *Protocol Data Unit*

RFC – *Request for Comments*

SFMP – *Simple Fixed Message Protocol*

SIT – *Sistemas Inteligentes de Transporte*

SMI – *Structure of Management Information*

SNMP – *Simple Network Management Protocol*

STMP – *Simple Transportation Management Protocol*

TCP – *Transmission Control Protocol*

TLV – *Tag-Length-Value*

UDP – *User Datagram Protocol*

1. INTRODUÇÃO

1.1. Considerações gerais

A dificuldade de monitoramento e gerência das vias de tráfego urbano, associadas ao crescente número de veículos, tem provocado um aumento significativo nos tempos e custos de viagem. A solução habitual para o problema é a expansão física das vias, que apesar de produzir um efeito positivo e imediato sobre a capacidade destas, não assegura por si só a melhoria da qualidade dos serviços. Além disso, a expansão de uma via acaba por utilizar espaços que poderiam ser ocupados por outras estruturas como prédios, áreas verdes ou de lazer, degradando a qualidade de vida dos habitantes nos centros urbanos.

Diante destas questões, diversos países implementaram estratégias de gerenciamento de tráfego, tendo como objetivo a otimização do uso dos recursos das vias. Uma das primeiras e mais simples destas estratégias surgiu com a instalação de semáforos nas interseções de vias públicas; apesar desta medida remontar ao início do século passado, continua sendo utilizada até hoje, permanecendo como uma das soluções mais inteligentes e simples para controle do fluxo de veículos (Homburger et al., 1992).

Ao longo de décadas, a demanda do tráfego fez com que este equipamento sofresse modificações e melhorias, tornando-se mais flexível e reativo ao tráfego (Homburger et al., 1992). Concomitantemente, avanços tecnológicos foram implementados em outros dispositivos, tais como sensores, contadores e detectores de veículos. Estes avanços, aliados ao surgimento de meios de comunicação mais eficientes, permitiram a integração destes dispositivos em um único sistema. Assim, foi possível migrar do controle automático de dispositivos para a gerência integrada dos sistemas de transporte, aumentando o foco das aplicações, o que permitiu então a construção de modelos de Sistemas Inteligentes de Transporte – SIT (FHWA, 1998).

Os modelos de SIT rompem o paradigma *veículo-via* para implantar o conceito de integração usuário-via-veículo, onde uma série de serviços automatizados são implementados e disponibilizados aos usuários, sejam motoristas, transportadores, passageiros ou pedestres, no intuito de melhorar o fluxo, minimizar custos e aumentar a segurança. Alguns destes serviços podem ser oferecidos junto à via em painéis eletrônicos, através de mensagens sobre as condições da pista ou em computadores localizados nos escritórios, para o monitoramento de cargas via Internet, entre outros.

Enquanto o conceito de SIT sedimentava-se, inúmeros produtos e serviços foram desenvolvidos e implantados sem um critério de padronização e em geral com características proprietárias (FHWA, 1999d). A implantação de dispositivos ou sistemas com estas características

acaba por gerar dois possíveis cenários: (a) ou os administradores de sistemas viários submetem-se às políticas de um único fornecedor e mantêm a conectividade entre dispositivos e sistemas, (b) ou constroem ilhas de SIT dentro de um mesmo sistema viário, utilizando produtos de fornecedores diferentes que não permitem a troca automática de dados entre si.

Foi buscando uma solução para as questões ligadas a conectividade e compatibilidade entre dispositivos e sistemas para SIT que várias agências dos Estados Unidos da América, em conjunto com a iniciativa privada, definiram uma arquitetura para produtos e serviços SIT, a *National ITS Architecture* -NIA (FHWA, 1999d).

A NIA descreve em sua proposta dois conceitos que considera fundamentais em sua arquitetura, o de interoperabilidade e o de intercambialidade. A interoperabilidade define que dispositivos e sistemas devem operar conjuntamente em um mesmo SIT, independentemente de fabricante ou de suas características funcionais internas. Já a intercambialidade tem pretensões mais arrojadas na concepção da NIA, seria possível a substituição de qualquer parte de um dispositivo ou sistema, por outro de um fornecedor diferente, sem a perda de funcionalidade ou estabilidade do SIT.

Não se pode dizer que a absorção destes conceitos pelos fornecedores de produtos SIT se deva unicamente à sua boa vontade. Oferecer produtos e serviços com estas características implica certamente em um aumento na concorrência e possível redução de lucros. Seguramente, o fator decisivo para a materialização destes conceitos foi e continua sendo a pressão exercida pelas agências e órgãos reguladores do tráfego nos Estados Unidos da América, que passaram a exigir em suas licitações produtos e serviços dentro de seu padrão de comunicação para SIT.

Porém, a intercambialidade, mesmo sendo muito interessante para planejadores de SIT, esbarra em problemas de ordem política e econômica, principalmente ligados à patente de produtos, inviabilizando sua implementação neste momento (FHWA, 1999d). Dentre todas as questões anteriores, a mais significativa é a de ordem tecnológica. Pois, se para garantir interoperabilidade basta conformidade tecnológica na comunicação, a intercambialidade exige a padronização interna dos dispositivos. Para isso, seria necessário que todos os fornecedores de produtos SIT tornassem públicas suas tecnologias. Isto favoreceria a concorrência, reduziria os lucros e tornaria economicamente inviável a pesquisa privada de novas tecnologias. Por estes e outros motivos, a intercambialidade ficou relegada a um segundo plano dentro da proposta da NIA.

1.2. Objetivos

1.2.1. Objetivo Geral

Este trabalho tem como objetivo geral estudar as tecnologias para troca de dados entre dispositivos SIT, através de um estudo de caso, onde uma das propostas de protocolo de comunicação padrão, seja implementada e avaliada. Mais especificamente, o estudo enfoca o conjunto de padrões de comunicação definidos para sistemas de gerência de tráfego, proposto na NIA, o *National Transportation Communications for ITS Protocol* – NTCIP. Este protocolo propõe-se a oferecer suporte para a troca de informações remotas, através de uma rede, entre centrais de controle de tráfego e sua respectiva instrumentação junto à pista.

1.2.2. Objetivos Específicos

Os objetivos específicos deste trabalho são:

- Implementar um software com as funcionalidades de comunicação definidas na proposta da *National Transportation Communications for ITS Protocol* - NTCIP para uma central de gerência de tráfego e sua respectiva instrumentação;
- Reconhecer e avaliar as questões tecnológicas envolvidas para aplicações reais deste protocolo;
- Desenvolver uma simulação onde possam ser avaliadas e compreendidas as características operacionais do protocolo.

1.3. Estrutura do trabalho

Durante os estudos preliminares realizados neste trabalho, surgiram inúmeras questões e a necessidade da compreensão dos diversos padrões de comunicação e gerência de informação, usados pelo *National Transportation Communication for ITS Protocol* - NTCIP. A avaliação e compreensão destes padrões mostraram-se fundamentais para a correta implementação do protocolo NTCIP, além de fornecer os fundamentos teóricos que permitiram definir quais as tecnologias que melhor atenderiam este objetivo.

Assim, para melhor descrever todos os aspectos tecnológicos envolvidos na implementação do NTCIP, este trabalho foi dividido em seis capítulos. O primeiro capítulo faz uma introdução ao tema e apresenta os objetivos a serem atingidos.

O capítulo 2 apresenta o conceito de SIT, seus principais sistemas e funcionalidades, os tipos de controle de tráfego atualmente implementados e uma breve descrição sobre arquiteturas de referência. No capítulo 3 tem-se uma visão abrangente da *National ITS Architecture*. É nesta arquitetura de referência para SIT que se insere o tema principal deste trabalho: o protocolo NTCIP.

No capítulo 4 estão os detalhes sobre as principais características, restrições e abrangências do NTCIP. Um estudo de caso é realizado no capítulo 5, onde os passos necessários para a implementação de uma aplicação NTCIP são detalhadamente descritos.

Finalmente, no capítulo 6 estão as conclusões à cerca dos estudos sobre a implementação e simulação de comunicação entre uma central de controle de tráfego e sua instrumentação junto à pista, utilizando um dos protocolos propostos no NTCIP.

2. SISTEMAS INTELIGENTES DE TRANSPORTE

2.1 Introdução

O aumento constante das demandas por transporte, independentemente do meio (ônibus, automóvel, trem e outros) tem sobrecarregado as vias de tráfego. Além disso, o crescente número de trocas modais, ou seja, de um meio de transporte para outro, tem exigido cada vez mais sincronia entre estes.

Para atender a estas demandas, diversos países, entre eles o Brasil, têm investido na tecnologia dos Sistemas Inteligentes de Transporte – SIT, na tentativa de minorar custos e melhorar a qualidade dos serviços. Estes sistemas caracterizam-se pelo uso maciço de tecnologias de informação e recursos de hardware e software, com a finalidade de integrar a infra-estrutura aos usuários da via.

Um SIT é formado por uma coleção de sistemas e serviços que tratam da gerência do tráfego e transporte, seja de cargas ou passageiros, oferecendo ferramentas para o controle de incidentes, estacionamentos, cobrança automática de taxas, trocas modais e outros.

Atualmente existem inúmeros fornecedores e produtos disponíveis no mercado, cada um definindo-se como “solução tecnológica”, fato que tem levado alguns órgãos responsáveis pela implantação deste tipo de serviço a investimentos equivocados.

A razão dos problemas de implantação, manutenção ou ampliação de um SIT está na falta de planejamento e conhecimento das tecnologias existentes no mercado. A seção 2.2 descreve algumas das principais características e serviços que podem ser implementados em um SIT.

2.2. Estrutura Operacional dos SIT

Existem vários sistemas passíveis de implementação em um SIT, porém alguns, por sua relevância e generalidade, estão presentes na maioria dos modelos descritos, são eles: Sistemas de informações ao viajante, gerenciamento do transporte público e de cargas, gestão de estacionamentos e tráfego.

2.2.1. Sistema de Informação ao Viajante

O sistema de informação ao viajante não se restringe aos motoristas, mas atende às mais diversas categorias de usuários, tais como, pedestres, ciclistas ou deficientes físicos. Sua função genérica é fornecer a cada usuário informações pertinentes às suas necessidades, a respeito das condições da via, da situação do tráfego, da infra-estrutura disponível, auxiliando na tomada de

decisão, melhorando o fluxo e permitindo o planejamento de itinerário por empresas de transporte de mercadorias, passageiros ou veículos particulares.

Várias são as aplicações que podem ser geradas com as informações distribuídas por este sistema:

- Planejamento de viagens;
- Estado de vias e tráfego;
- Serviços disponíveis na via;
- Comunicação pessoal;
- Algoritmo de otimização de rotas.

2.2.2. Sistema de Gerenciamento do Transporte Público

Os SIT têm um grande potencial neste campo, pois contribuem de diversas formas no dimensionamento, controle e fluxo dos transportes públicos. A partir da coleta de informações sobre a via, veículos e passageiros, pode-se então interferir no tráfego, priorizando o fluxo deste tipo de transporte. Além disso, os dados armazenados proporcionam uma fonte de pesquisa para o dimensionamento de frotas, grades de horário, previsão de demandas e tipos de transporte mais utilizados.

Estes dados podem ser coletados e distribuídos, utilizando-se diversos tipos de equipamentos como catracas eletrônicas, terminais interativos, monitoramento automático de veículo e outros. As aplicações mais comuns para as informações coletadas são:

- Elaboração de escalas de horário;
- Gerenciamento de operação;
- Serviço de informação ao público;
- Controle automático de passageiros.

2.2.3. Sistema de Gerenciamento do Transporte de Cargas

O setor de transporte de cargas oferece um campo para inúmeras aplicações de SIT, desde o planejamento de rotas e monitoramento da posição geográfica do veículo, à situação da carga.

Algumas aplicações SIT de menor escala já estão em utilização em diversos países. Por exemplo, tem-se o *Automatic Vehicle Location* – AVL que prevê a localização automática de

veículos integrada aos recursos da Internet, disponibilizando a transportadores e usuários a localização do veículo e dados sobre a carga. Podemos destacar outras aplicações possíveis:

- Controle de operações;
- Controle de frotas;
- Monitoração de cargas.

2.2.4. Sistema de Gerenciamento de Estacionamento

Um veículo permanece apenas 10% de sua vida útil trafegando; o restante do tempo está estacionado. Desta forma, o gerenciamento de estacionamentos é muito importante dentro da otimização do sistema de transportes.

Este sistema pode ser aplicado tanto em vias públicas como em ambientes restritos. Sua principal vantagem é evitar que veículos fiquem rodando em busca de vagas, reduzindo atrasos, consumo de combustível e congestionamentos.

As principais aplicações disponibilizadas nesta área pelos Sistemas Inteligentes de Transporte são:

- Gerenciamento de espaços nos estacionamentos;
- Aplicações voltadas ao auxílio à direção;
- Informações sobre trocas modais;
- Aplicações voltadas à reserva de vagas;
- Sistema automático de pagamentos.

2.2.5. Sistema de Controle de Tráfego

Uma Central de Controle de Tráfego monitora em tempo real a via e veículos, sendo que as principais aplicações geradas são:

- Controle de tráfego;
- Gerenciamento de incidentes e prestação de socorro;
- Gerenciamento de acessos;
- Monitoração de condições ambientais;
- Restrições de demanda;
- Monitoração de veículo;
- Auxílio a direção.

Tradicionalmente, sistemas de controle de tráfego são implantados para minimizar atrasos, coordenar o fluxo de veículos e maximizar os recursos da via. Para isso, são utilizados inúmeros dispositivos, o que não significa necessariamente alta tecnologia. Frequentemente, o uso de semáforos nas interseções é suficiente para atingir grande parte das metas de controle sobre uma via.

A evolução dos equipamentos de controle semafórico tornou possível o desenvolvimento de estratégias de controle mais flexíveis e sofisticadas e por consequência mais eficiência na gerência do tráfego. É a coordenação individual ou conjunta destes dispositivos que garante a eficiência do sistema. Basicamente, existem dois tipos de controle de tráfego urbano que são atualmente aplicados a partir do uso de semáforos:

- **Controle por tempo fixo**

O controle por tempo fixo utiliza planos semafóricos pré-programados, calculados previamente. Estes planos determinam os tempos de verde e vermelho de cada semáforo.

Esta técnica de controle, apesar de sua funcionalidade e ampla utilização, possui algumas características indesejáveis, já que os dados utilizados para o cálculo destes planos alteram-se durante o decorrer do dia devido às variações do fluxo de veículos. Assim, diversos planos semafóricos devem ser calculados para diferentes horários, visando o ajuste dos tempos de verde em relação ao fluxo.

A coordenação dos tempos de verde não varia de ciclo para ciclo. Assim, sistemas atuam satisfatoriamente somente sobre modelos de tráfego conhecidos, não possuindo a capacidade de adequar-se a rápidas variações do fluxo.

Sistemas de tempo fixo são implementados de maneira a atender as mudanças de fluxo diários, semanais ou em casos especiais. Responder adequadamente à demanda do tráfego, determinando o momento exato da mudança dos planos é a maior dificuldade deste sistema.

Apesar dos benefícios oferecidos por este sistema, ele tende a degradar-se à medida que as condições do tráfego mudem. É estimado que em sistemas de tempo fixo, os planos semafóricos sofram uma degradação de 3% ao ano, e portanto em um prazo de cinco anos serem perdidos todos os benefícios iniciais (Hills et al., 1987). Alguns dos principais fatores de degradação de um sistema de tempos fixos, segundo Hills et al. (1987), são:

- Alterações na malha viária;
- Modificações no fluxo devido ao aumento da demanda;

- Alterações de fluxo em uma ou mais partes do sistema.

- **Controle atuado pelo tráfego**

É a realimentação do sistema que diferencia o controle atuado pelo tráfego do de tempo fixo. O controle atuado pelo tráfego utiliza-se de dispositivos de detecção junto à pista para dotar-se das informações necessárias e então, a partir destas, processar e determinar as ações, adequando gradualmente os tempos de verde às condições do tráfego naquele momento. Segundo Homburger et al., (1992), as principais vantagens da utilização de controle atuado pelo tráfego em relação ao de tempos fixos são:

- As alterações dos tempos não necessitam de dados históricos;
- Não existem mudanças repentinas de tempos;
- O sistema ajusta-se aos mais diversos eventos (shows, incidentes, etc.);
- Não há necessidade de planos pré-calculados;
- Pode-se atuar instantaneamente no tráfego para atender demandas de veículos especiais (bombeiros, ambulâncias, etc.).

Apesar das vantagens deste tipo de controle sobre os demais, existem alguns pontos considerados críticos no controle totalmente atuado. O sistema é muito sensível à falha dos detectores, a manutenção do sistema é crítica, tornando indispensável uma supervisão central do sistema e, finalmente, o custo de implantação é consideravelmente maior do que no anterior (Homburger et al., 1992).

2.3. Arquiteturas de Referência para SIT

A descrição dos SIT feita na seção 2.2, indica que estes sistemas são heterogêneos, distribuídos e complementares e que sua integração seria de grande valia na gerência da operação dos sistemas viários. Foi buscando maior eficiência de gerência nestes sistemas que programas para o transporte inteligente em diversos países, como Canadá, Estados Unidos da América, Japão, Austrália e Comunidade Européia, definiram suas arquiteturas de referência nacionais para SIT. Nesses países, a definição de uma arquitetura de referência nacional surgiu da iniciativa do setor público em cooperação com empresas privadas, permitindo a reestruturação de seus sistemas de transporte e retardando a imediata necessidade da expansão física das vias.

Uma arquitetura SIT é uma estrutura para o desenvolvimento de produtos e sistemas de transporte, onde estão definidas estratégias e limites. Não se trata de um projeto de sistema, mas de

um modelo sobre o qual diversas abordagens podem ser construídas, cada uma atendendo necessidades específicas sem que se perca a integração e interoperabilidade.

Segundo Bass et al. (1998), uma arquitetura de software é composta por estruturas que incluem componentes, suas propriedades externas e os relacionamentos entre eles, constituindo uma abstração do sistema. A abstração permite que detalhes que não afetem a operação dos componentes do sistema possam ser suprimidos, reduzindo a complexidade da descrição e auxiliando o gerenciamento desta.

A arquitetura constitui também um ponto de referência para as atividades que serão executadas após sua definição. Isto significa que a arquitetura é a manifestação antecipada das decisões do projeto, preocupando-se com o tempo de desenvolvimento, custo de manutenção, definição das restrições da implementação e da estrutura organizacional, enfatizando os atributos de qualidade que o sistema requer e medindo através de avaliações a empregabilidade das qualidades necessárias (Bass et al., 1998). Assim, uma arquitetura de referência consiste em um conjunto de componentes de software e os relacionamentos entre estes, de maneira que as funcionalidades descritas no modelo possam ser implementadas.

Em Jazayeri et al. (2000), uma arquitetura de software é uma ferramenta criada para lidar com a complexidade do software e esta deve satisfazer os requisitos funcionais e não funcionais do sistema. Além disso, uma arquitetura é uma interface de dois componentes distintos: o problema de negócio e a solução técnica deste (Hernán et al., 1998).

Sistemas Inteligentes de Transporte são um exemplo de ambiente no qual o nível de complexidade é elevado, devido ao grande número de serviços implementáveis conforme visto na seção anterior.

Apesar das várias arquiteturas de referência para SIT desenvolvidas ou em desenvolvimento no mundo, duas delas encontram-se melhor documentadas, com alguns sistemas implementados e em avançado grau de amadurecimento da proposta. São elas a *National ITS Architecture* - NIA e a *Keystone Architecture Required for European Networks* - KAREN, a primeira desenvolvida nos Estados Unidos da América e a segunda na Comunidade Européia.

2.3.1. O Modelo de Arquitetura para SIT Norte-Americano

A *National ITS Architecture* surgiu da necessidade de otimização dos recursos das vias de tráfego do sistema de transporte norte-americano, através do gerenciamento mais eficaz destes recursos. As primeiras pesquisas sobre este tema surgiram ainda na década de 60, juntamente com os sistemas automatizados de gerência de tráfego urbano, além de algumas aplicações para transporte rodoviário e aeroviário (FHWA, 1998). Porém, foi na década de 90, com a união de fornecedores de produtos para SIT e agências como

American Association of State Highway and Transportation Officials - AASHTO, Institute of Transportation Engineers - ITE e National Electrical Manufacturers Association - NEMA, que começaram a ser delineados os primeiros esboços para a estruturação de uma arquitetura de referência, que permitiu então, a construção de produtos com características de interoperabilidade. A partir deste momento, um grande número de documentos passa a ser disponibilizado ao público, no intuito de esclarecer a abrangência, as vantagens operacionais e mercadológicas oferecidas por esta arquitetura.

Apesar da *National ITS Architecture* permitir a gerência dos meios de transporte em geral, incluindo marítimo, aéreo, ferroviário e rodoviário, estes dois últimos são o foco principal das pesquisas e do desenvolvimento de produtos e assim, encontram-se melhor documentados.

A NIA fornece um conjunto de subsistemas módulos que pretende atender a todas as necessidades de gerência dos transportes. Porém, como todo SIT possui peculiaridades, a arquitetura também se ambienta ao sistema norte-americano de transporte, evocando, por exemplo, uma tendência à otimização do fluxo de veículos particulares, relegando a um segundo plano o transporte público de passageiros, ainda que este seja contemplado na proposta.

Apesar destas e outras características particulares da arquitetura, o conjunto da proposta é bastante genérico e não pretende ser um padrão definitivo ou acabado, mas uma estrutura aberta que evolui conjuntamente com as necessidades e os avanços tecnológicos da área de SIT (FHWA, 1999d).

Em 1995, um conjunto de padrões para o sistema de comunicação entre dispositivos SIT foi definido e publicado em um documento denominado NTCIP Guide 9001 pelas agências AASHTO, ITE e NEMA. Com isto, inúmeras pesquisas e produtos foram desenvolvidos, gerando a necessidade de alterações para adequar a arquitetura ao mundo real. As falhas encontradas na proposta da NIA levaram a sua revisão, principalmente na área de comunicação de dados; uma nova versão foi publicada em 1998. Outras revisões foram feitas; a última, em 2002, veio com o intuito de esclarecer aos fornecedores de produtos SIT como proceder para padronização de seus equipamentos.

Como foi dito, a *National ITS Architecture* não é uma proposta definitiva, mas um modelo que surgiu e evolui com as demandas do sistema de transporte e sua função é de determinar apenas os limites tecnológicos para produtos e serviços, garantindo a padronização dos Sistemas Inteligentes de Transporte. As principais características e funções desta arquitetura são alvo deste trabalho e estão descritas no Capítulo 3.

2.3.2. O Modelo Europeu de Arquitetura SIT

A arquitetura de referência para implementação de Sistemas Inteligentes de Transporte proposto pela Comunidade Européia denominada, *Keystone Architecture Required for European Networks* - KAREN, possui um perfil semelhante ao modelo norte-americano.

As diferenças entre as duas residem na forma de estruturação da arquitetura. Enquanto a *National ITS Architecture* busca prioritariamente a otimização do fluxo de tráfego ao menor custo possível, o modelo dado pela KAREN prioriza a interconectividade de sistemas, trocas multimodais, transporte público de passageiros e a segurança viária em um ambiente multinacional.

Apesar de existirem sistemas em operação que seguem a arquitetura referência para SIT KAREN em Turim, Milão e outras cidades européias (TAP, 1999), esta se encontra ainda em fase de amadurecimento. Além disso, o número de publicações e a documentação em geral é significativamente menor com relação a *National ITS Architecture*.

As principais características desta arquitetura são: oferecer um conjunto de ferramentas para o intenso monitoramento das vias; prover informações aos viajantes internacionais; permitir a gerência de trocas modais, seja para o transporte de mercadorias ou de passageiros; e garantir um bom nível de segurança aos usuários do sistema. A Figura 2.1 ilustra o sistema de controle de tráfego urbano e transporte público da arquitetura KAREN, na forma de diagrama de contexto.

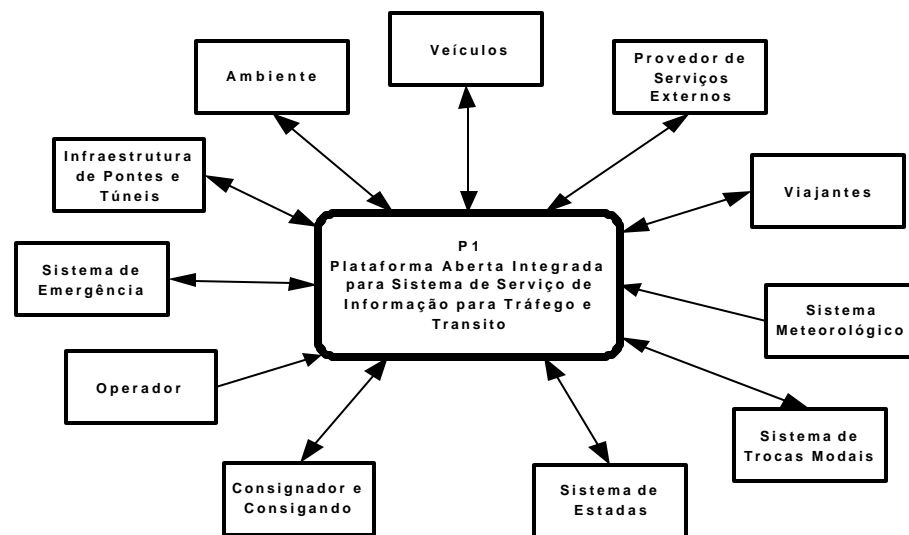


Figura 2. 1– Diagrama de Contexto do Sistema de Gerenciamento de Transporte Público e Tráfego Urbano (TAP, 1999).

2.4. Considerações Finais

Com o crescente aumento dos volumes de tráfego, os SIT vem ganhando espaço junto aos gestores do transporte em todos os países, como uma das formas de melhorar a qualidade dos serviços sem o ônus da ampliação física das vias. A utilização desta nova tecnologia, que tem como matéria prima a informação e como produto sua disseminação, vem ao encontro das necessidades dos usuários dos sistemas de transporte, que reclamam por maior eficiência e segurança.

A implementação de um SIT tem como prerrogativa melhorar a qualidade dos serviços através do aumento da eficiência da via, otimizando recursos, reduzindo tempos e custos de viagem, além prover mecanismos para a segurança dos usuários.

O aumento das aplicações implantadas sem um critério norteador acabou por gerar inúmeros sistemas que apesar de eficientes não possuem conectividade entre si. Esta incapacidade de interoperação, criada pelo uso de tecnologias proprietárias, levou à definição de arquiteturas de referência para SIT em diversos países. Apesar das peculiaridades de cada uma destas arquiteturas, há uma convergência para que a interoperabilidade entre sistemas e dispositivos SIT venha a concretizar -se em um futuro próximo através do uso de tecnologias e padrões abertos.

3. A PROPOSTA DE ARQUITETURA DE REFERÊNCIA DA NATIONAL ITS ARCHITECTURE - NIA

3.1. Introdução

Conforme visto na seção 2.3.1, uma arquitetura de referência foi definida nos EUA para nortear o desenvolvimento de Sistemas Inteligentes de Transporte. Esta arquitetura será adotada neste trabalho por ser a que fornece a visão mais detalhada dos serviços a serem prestados pelos SIT e, também, aquela com melhor documentação disponível para consulta.

A estrutura proposta pela *National ITS Architecture* permite planejar, definir e integrar Sistemas Inteligentes de Transporte. Suas principais características são: a baixa dependência existente entre subsistemas e módulos, o uso de tecnologias reconhecidas pela indústria e requisitos mínimos de comunicação baseados em padrões abertos (FHWA, 1999a).

Estas características permitem a implantação parcial de sistemas ou subsistemas, com a garantia de que a estabilidade do modelo não se perca, adaptando assim, a arquitetura às necessidades e possibilidades do ambiente. A exigência básica da arquitetura é que todos os subsistemas e módulos mantenham a característica da interoperabilidade independentemente das alterações tecnológicas dos dispositivos e dos meios de comunicação (FHWA, 2002a).

A proposta da NIA subdivide-se em dois níveis conceituais, denominados arquitetura lógica e arquitetura física. Ambos os níveis derivam das demandas geradas pelo ambiente de tráfego, ou seja, das solicitações de serviços e informações dos usuários da arquitetura. A arquitetura lógica define as funções do sistema, tais como detecção de incidentes, controle de rampas de acesso e outros. Em outro nível, a arquitetura física define onde serão implementadas as funções definidas na arquitetura lógica, como ilustra a Figura 3.1.

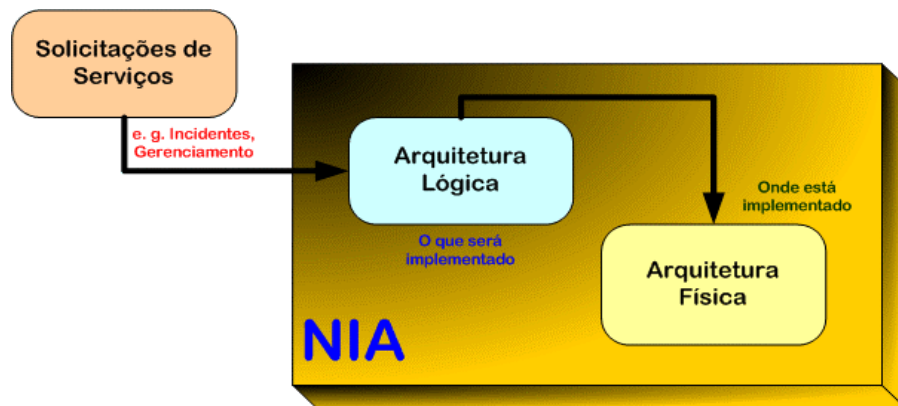


Figura 3. 1 – Simplificação da NIA.

3.2. Arquitetura Lógica

A arquitetura lógica descreve o fluxo das informações dentro de um sistema inteligente de transporte, desde o mais alto nível do sistema até o mais baixo, ou seja, das funções a serem implementadas às informações que poderão ser trocadas entre estas funções. Além disso, oferece uma visão dos serviços que serão implementados e disponibilizados aos usuários, sem levar em conta os instrumentos necessários para que isto ocorra.

As funções a serem implementadas são os processos ou sistemas da arquitetura, enquanto as informações são os dados necessários para a viabilização do SIT (FHWA, 1999a). A Figura 3.2, ilustra de forma simplificada o diagrama em alto nível da Arquitetura Lógica.

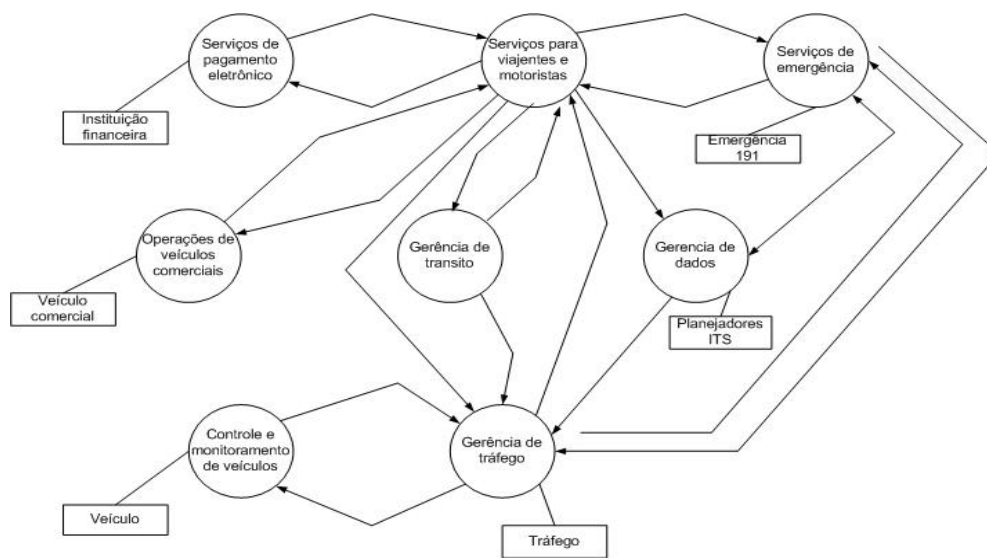


Figura 3.2 - Descrição em Alto Nível da Arquitetura Lógica (FHWA, 1999c).

3.3. Arquitetura Física

A arquitetura física provê o suporte para as definições construídas na arquitetura lógica. É neste nível que estão descritos os padrões, cenários e considerações sobre os componentes físicos do sistema, que servirão de base para a implementação dos módulos da arquitetura.

Basicamente, a NIA fundamenta-se sobre as interações de três camadas de infraestrutura: uma camada de transporte, uma de comunicação e outra institucional, como ilustra a Figura 3.3.

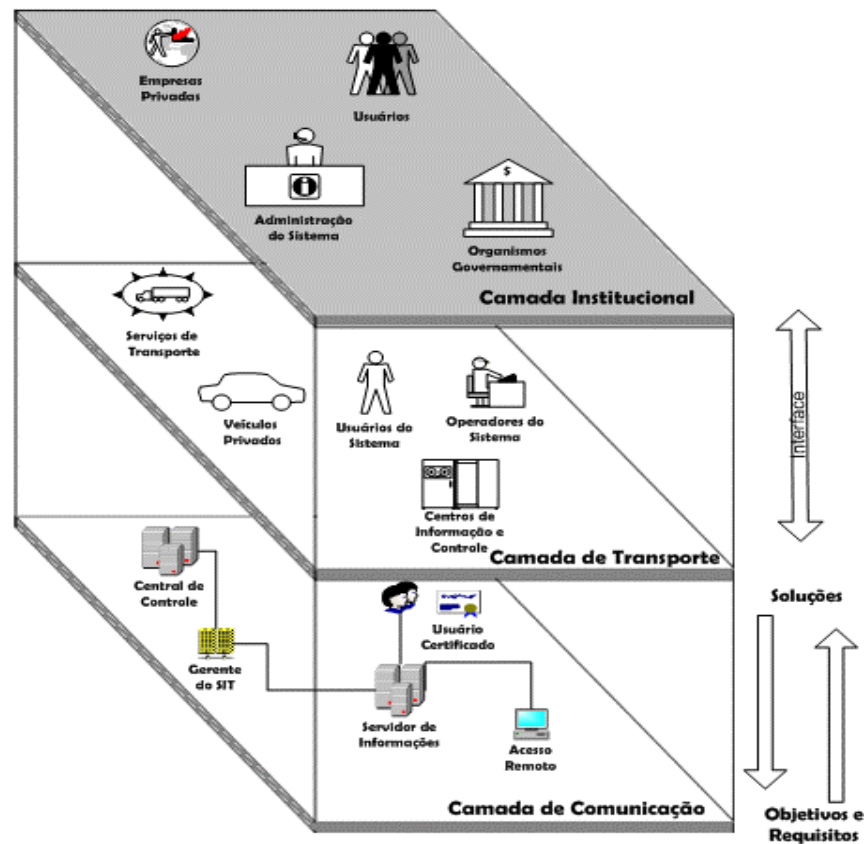


Figura 3.3 - Camadas da Arquitetura Física da *National ITS Architecture* (FHWA, 1999b).

3.3.1. Camada Institucional

A camada institucional representa os mentores do projeto, planejadores e usuários do Sistema Inteligente de Transporte. Nesta camada, são definidas as estratégias que serão utilizadas e quais os subsistema e módulos a serem implementados no SIT. Além disso, é nesta camada que estão os responsáveis pelo suporte, gerenciamento e controle do SIT (FHWA, 1999b).

3.3.2 Camada de Comunicação

Na camada de comunicação estão descritas as tecnologias que proverão as interfaces entre as funções da camada de transporte, os tipos de dados e os serviços permitidos para troca de informações (FHWA, 1999b). Esta é a camada que proporciona a coordenação e o compartilhamento de informações entre o sistema e os usuários.

Os recursos de comunicação tecnologicamente possíveis em um SIT são todos os que puderem atender a demanda do fluxo de dados e os requisitos de tempo desejados. A proposta da

NIA é a de que as redes para SIT não sejam necessariamente exclusivas, mas que permitam outros serviços compartilhados, como ilustra a Figura 3.4.

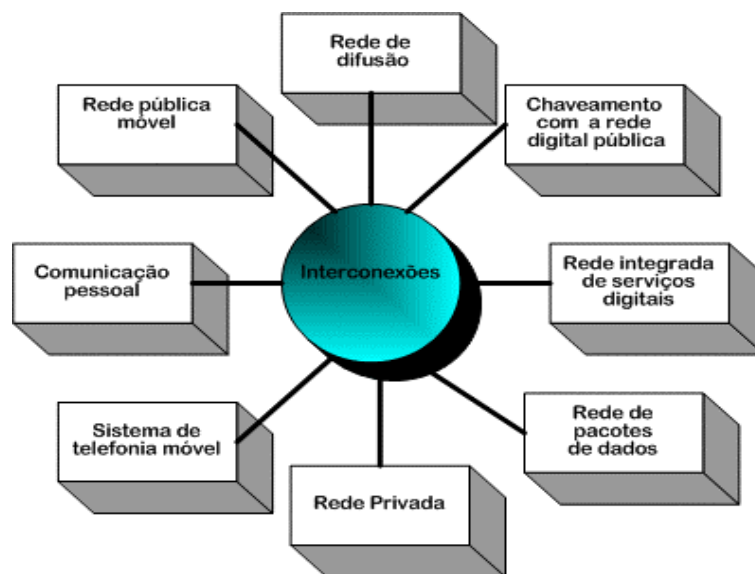


Figura 3.4 - Visão geral dos serviços de telecomunicação para *National ITS Architecture* (FHWA, 1999).

A flexibilização dos meios físicos de comunicação proposta na *National ITS Architecture* tem uma função específica, permitir que diversos sistemas espalhados geograficamente possam utilizar-se das redes públicas com tecnologia Internet (FHWA, 1997a). Esta característica permite que a rede possa expandir-se a qualquer tempo sem a necessidade de grandes investimentos na arquitetura de comunicação.

Atualmente, grande parte dos SIT possui sistemas de comunicação dedicados, utilizando fibra óptica para o controle de tráfego e vigilância ou rádio, para gerenciamento de cargas e frotas. O crescimento destas redes, seu amadurecimento e interconexão, permitirão que os SIT as compartilhem, reduzindo o custo de implantação, viabilizando assim a expansão e instalação de novos sistemas.

3.3.3. Camada de Transporte

A camada de transporte executa as funções especificadas da arquitetura lógica, servindo como interface da as informações colhidas e processadas no SIT e os usuários do sistema. Nesta camada são executados os controles sobre a informação, restringindo ou liberando um canal de comunicação a um usuário ou subsistema, dependendo de seu nível de acesso.

3.4. Subsistemas da Arquitetura Física

Um subsistema encapsula um conjunto de módulos, cada qual com suas funções, sendo que estas funções são aquelas descritas na arquitetura lógica e aqui mapeadas. A *National ITS Architecture* possui vinte e um módulos, divididos em quatro subsistemas, interconectados através de canais de comunicação, como ilustra a Figura 3.5.

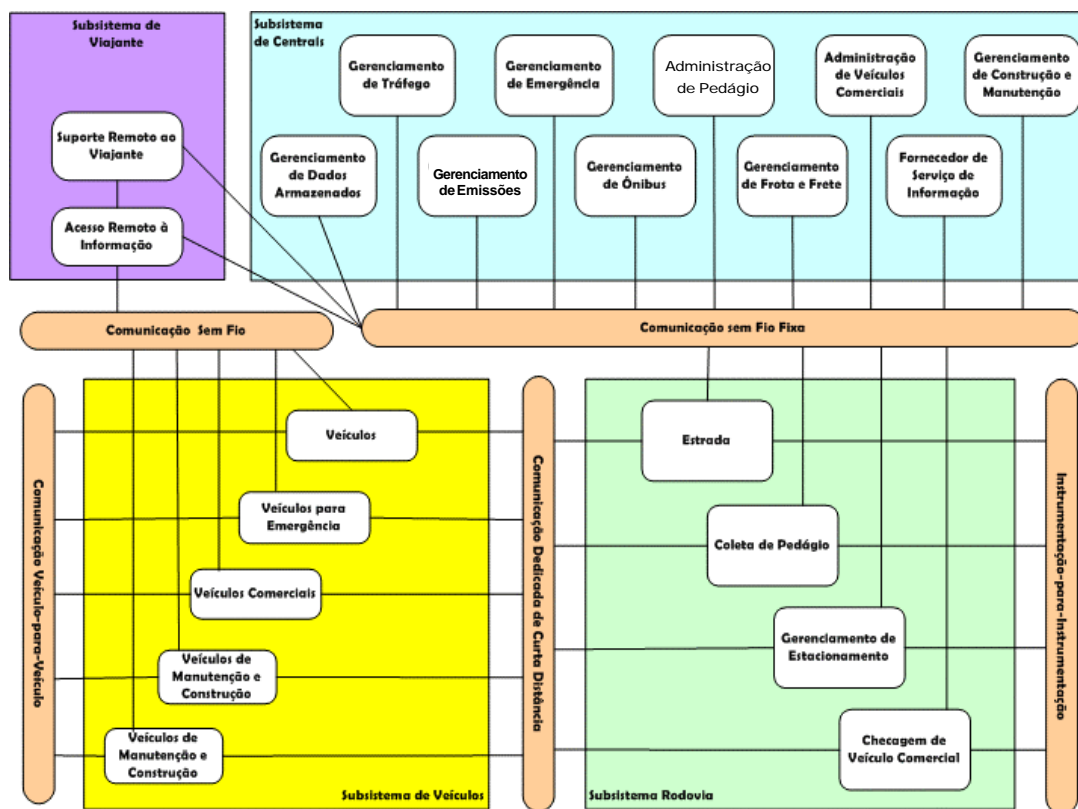


Figura 3.5 - Subsistemas e Comunicação da *National ITS Architecture* (FHWA, 1999d).

Cada subsistema definido dentro da NIA é formado por um conjunto de módulos, agrupados por similaridades de funções. São as funções executadas por cada um destes módulos que tornam um SIT efetivamente operacional. São quatro os subsistemas definidos dentro da proposta de arquitetura da NIA: Veículos, Rodovias, Centrais e Viajantes.

3.4.1. Subistema de Centrais

O subsistema de centrais da NIA propõe que cada central compartilhe seus dados, recursos de comunicação e instrumentação de beira de pista, mas que mantenham independência funcional entre si. Não existem pré-requisitos que indiquem que os módulos devam estar adjacentes ou sobrepostos na implantação. Esta característica flexibiliza a arquitetura e permite que sejam

implantados somente os módulos necessários ao ambiente. O subsistema de centrais subdivide-se em dez módulos:

- **Administração de veículos comerciais:** Negocia e administra taxas e credenciais, guarda os registros de segurança, informando a outras centrais sobre os veículos comerciais que se encontram ou encontravam-se em sua região.
- **Gerenciamento de frota e fretes:** Coordena frotas e trocas intermodais.
- **Administração de taxas:** Executa tarefa de cobrança automática de pedágio.
- **Gerenciamento de ônibus:** Coleta dados operacionais dos veículos em trânsito e executa estratégias e planos táticos para veículos e motoristas.
- **Gerenciamento de emergências:** Coordena a resposta a incidentes, inclusive os com materiais perigosos.
- **Gerenciamento de emissões:** Coleta informações referentes à poluição e provê a entrada de dados para o *Traffic Management*.
- **Gerenciamento de dados armazenados:** Responsável pelo armazenamento e distribuição interna dos dados coletados no SIT.
- **Gerenciamento de tráfego:** Processa dados e fornece informações básicas para o tráfego em condições normais ou em caso de incidentes.
- **Fornecedor de serviço de informações:** Coleta processa e distribui dados do SIT para usuários externos (viajantes, empresas, etc.) na forma padrão ou personalizada.

- **Gerenciamento de Construção e Manutenção:** Fornece controles para redução de impacto sobre o tráfego no caso de manutenção ou construção de vias.

3.4.2. Subsistema de Rodovias

Este subsistema é composto por um conjunto de dispositivos tais como sensores, sinais luminosos, painéis programáveis ou outras interfaces instaladas junto à pista. Estes dispositivos podem prover de dados tanto as centrais quanto os veículos que estiverem ao longo da via. O subsistema é dividido em quatro módulos segundo suas funções:

- **Estrada:** Executa o gerenciamento de tráfego, coletando dados e atuando sobre os dispositivos de beira de pista. Além disso, envia sinais para os veículos e outros usuários da via.
- **Coleta de Pedágio:** Interage com os veículos coletando taxas de pedágio e identificando violações.
- **Gerenciamento de Estacionamentos:** Coleta taxas de estacionamento, além de dados de ocupação, disponibilizando estas informações aos viajantes.
- **Conferência de Veículos Comerciais:** Verifica as credencias e dados dos veículos, além de determinar padrões e requisitos mínimos de segurança. Envia dados para o motorista e em alguns casos para o transportador.

3.4.3. Subsistema de Veículos

Neste subsistema os módulos de funções estão integrados aos veículos e são responsáveis pelas informações disponibilizadas ao motorista, tais como, navegação do veículo e funções de segurança. Os módulos do subsistema comunicam-se com as centrais através da instrumentação instalada junto à pista. As aplicações previstas pela NIA para cada tipo de veículo neste subsistema são:

- **Veículos:** Provê serviços de coleta e processamento de informações para garantir a segurança do veículo. Além disso, fornece acesso a

informações sobre rotas, controle do veículo, aviso de pré-acidente e outros.

- **Veículos de Emergência:** Fornece a localização de serviços de emergência, plano de rotas e outros.
- **Veículos Comerciais:** Provê a comunicação entre a instrumentação da via e o veículo e deste com o transportador. Fornece também o serviço de coleta e avaliação de estado do veículo, motorista e carga, permitindo a emissão de um sinal de alerta, toda vez que há algum problema de segurança.
- **Veículos de Manutenção e Construção:** Auxilia na coleta e processamento de dados para garantir a segurança e o fluxo de veículos, em caso de bloqueios da pista para manutenção ou construção.
- **Transporte Público:** Neste módulo diversos serviços são disponibilizados, como cobrança eletrônica de passagem, priorização ou retardamento de um veículo, fornecimento de rotas e horários de serviços.

3.4.4. Subsistema de Viajantes

Fornece as informações contidas no SIT a seus usuários, liberando o acesso ao sistema dentro do nível de permissão dado a cada um deles. O subsistema possui dois módulos definidos na arquitetura:

- **Suporte Remoto a Viajantes:** Provê dados a usuários que façam acesso ao SIT.
- **Acesso Particular de Informação:** Fornece serviços especiais para usuários do SIT, tais como, transações financeiras ou comunicação particular de dados.

Os subsistemas e seus módulos, juntamente com os meios de comunicação formam a *National ITS Architecture*. Isto não significa que todos os itens descritos na proposta devam ser implantados. O que a arquitetura oferece é um intervalo de funções e padrões para a implementação de uma aplicação SIT.

3.5. Considerações Finais

A *National ITS Architecture* fornece uma estrutura comum a todos os desenvolvedores para que estes possam planejar, definir e integrar seus SIT. A arquitetura não determina quando ou como deve ser implementado um subsistema ou módulo, mas descreve quais os serviços que o subsistema deve fornecer e em que módulo devem estar estes serviços.

A função da arquitetura é de manter a interoperabilidade entre os subsistemas e módulos, garantindo a conectividade interna e externa do sistema, além de sua estabilidade. A *National ITS Architecture* firma um padrão, sem restringir as características individuais de cada sistema, permitindo assim uma grande flexibilidade na implantação e expansão das aplicações SIT.

Na arquitetura são definidos os subsistemas de seus respectivos módulos funcionais, mas é a infraestrutura de comunicação que efetivamente permite a interoperabilidade dos SIT. A infraestrutura de comunicação na NIA não se restringe a parte física de suporte a troca de dados, mas ao conjunto de tecnologias e padrões usados para implementá-la. É o reconhecimento destes padrões de hardware e software que permite aos desenvolvedores e planejadores de SIT, efetivamente implementar produtos e serviços com as características de interoperabilidade e intercambialidade desejadas.

A NIA utiliza um conjunto de padrões de comunicação, armazenamento e codificação de informação, baseado em tecnologias abertas e reconhecidas pela indústria. No Anexo 1, estão descritos os principais padrões e tecnologias de comunicação da NIA, utilizados dentro de seu protocolo de comunicação, denominado *National Transportation Communications for ITS Protocol* – *NTCIP*.

4. A INFRAESTRUTURA DE COMUNICAÇÃO DO NTCIP

4.1. Introdução

O *National Transportation Communications for ITS Protocol* - NTCIP é formado por um conjunto de padrões que descrevem as regras de comunicação (protocolos) e o vocabulário (objetos de dados) necessário, para que possa existir a troca de informações entre dispositivos SIT desenvolvidos por diferentes fabricantes.

As aplicações com NTCIP utilizam conceitos como ASN.1, MIB, SNMP e BER, abordados no Anexo 1. A apresentação deste capítulo pressupõe familiaridade com os conceitos ali descritos. Apesar destes padrões já serem velhos conhecidos do ambiente Internet, sua aplicação no âmbito dos SIT é recente e inovadora. E, portanto, a compreensão do funcionamento, alcance e restrições das tecnologias citadas anteriormente, são fundamentais para o entendimento e implementação do NTCIP.

O NTCIP é uma proposta pioneira no desenvolvimento de uma infraestrutura de comunicação para dispositivos de controle de tráfego, construída de forma a permitir que dispositivos de diferentes fabricantes venham a ser usados sobre um mesmo canal de comunicação. A Figura 4.1 descreve a estrutura geral de comunicação fornecida pelo NTCIP sobre a arquitetura NIA.

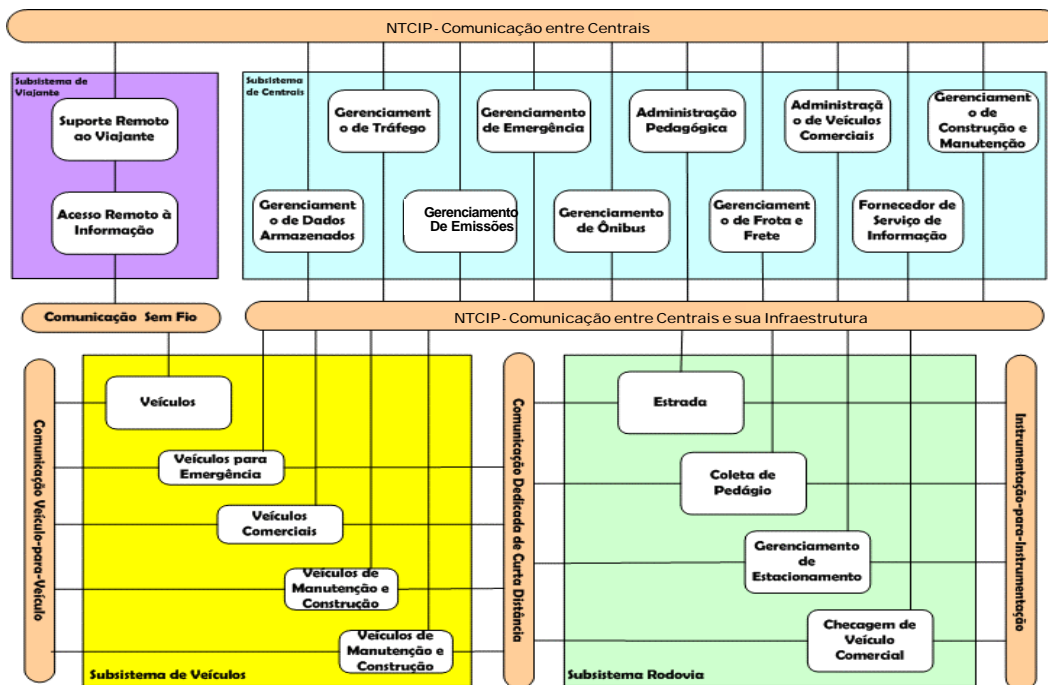


Figura 4.1 – O NTCIP e a *National ITS Architecture* (FHWA, 1999d).

Até este momento, a proposta do NTCIP para comunicação entre dispositivos SIT divide-se em dois grupos: um conjunto de padrões para a troca de informação entre centrais (*Center-to-Center Communications*) e outro para o gerenciamento de dispositivos de campo (*Center-to-Field Communications*).

Entre centrais, o tipo de comunicação é semelhante à Internet, onde qualquer central pode requisitar ou fornecer informações (FHWA, 1999d). Apesar das centrais operarem com grande volume de informações, gerados por sua instrumentação ou armazenadas em seus bancos de dados, suas restrições temporais são menos rígidas, permitindo o uso de padrões de comunicação genéricos como DATEX, CORBA ou FTP.

Já para comunicação entre as centrais e sua instrumentação junto à pista, o uso de protocolos com características tão abrangentes quanto no caso anterior é restrito. A proposta do NTCIP para este grupo de comunicação faz uso de um protocolo genérico, o SNMP. Além deste, outros foram desenvolvidos especificamente para esta função, como o *Simple Network Management Protocol* – STMP e o *Simple Fixed Message Protocol* - SFMP. Estes protocolos pretendem tanto oferecer facilidades no desenvolvimento e gerência do SIT, como atender demandas especiais de tempo real, grandes fluxos e baixos volumes de informação por pacote de dados. Na Figura 4.2, estão descritos os padrões e tecnologias usados para interconectar subsistemas e módulos usando o NTCIP.

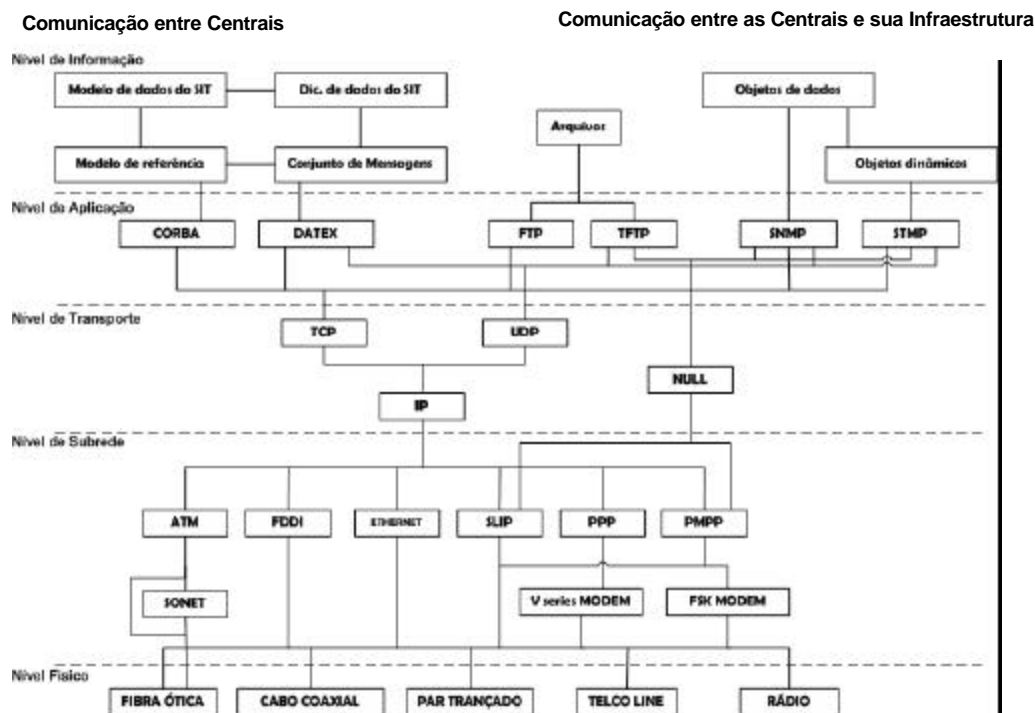


Figura 4.2. – Conjunto de padrões de comunicação do NTCIP (FHWA, 1999d).

O NTCIP não define os detalhes de como as centrais ou dispositivos deverão operar, ou que tipo de tecnologia pode ou não ser utilizada em cada um. Porém, especifica claramente os critérios de descrição, armazenamento, codificação, transmissão e decodificação dos dados que podem ser trocados entre centrais e dispositivos que estão se comunicando. Para isto, o NTCIP subdivide-se em cinco níveis operacionais, de forma similar às camadas do modelo OSI/ISO, onde cada nível descreve um grupo de funções, padrões ou tecnologias permitidas.

4.2. Níveis Operacionais do NTCIP

O NTCIP foi projetado inicialmente para suportar a comunicação entre centrais de gerenciamento de tráfego e controladores semafóricos, por esta ser vista como a área de maior urgência para a implantação de um protocolo padrão (FHWA, 2002a). Entretanto, durante o processo de desenvolvimento da proposta, esta se estendeu para outros subsistemas SIT. A perspectiva dos desenvolvedores é que este venha expandir-se até cobrir todas as instâncias de comunicação entre os diversos dispositivos de tráfego (NEMA, 1996).

Originalmente, o NTCIP foi definido sobre um conjunto de perfis; a classificação de cada um destes estava relacionada ao tipo de comunicação e tecnologias utilizadas. Um perfil em particular definia uma combinação de objetos e protocolos de uso comum entre um grupo de dispositivos. Por exemplo, usando-se um canal de alta velocidade para a comunicação entre centrais e seus controladores semafóricos, se estaria sob o perfil A. Porém, no mesmo exemplo, mas usando um canal de baixa velocidade (300 ou 1200 bauds), o perfil seria denominado B. Esta idéia foi substituída mais tarde pelo conceito de pilha de protocolos e atualmente o NTCIP define um conjunto de padrões distribuídos em níveis, como ilustra a Figura 4.2. Nestes níveis estão descritos os objetos, padrões e tecnologias que serão usados para a construção da estrutura de comunicação entre os diversos dispositivos do SIT. O NTCIP busca no modelo OSI/ISO uma referência para sua estrutura de protocolo. Apesar do modelo OSI/ISO referir-se às camadas operacionais de um protocolo e o NTCIP definir um conjunto de padrões distribuídos em níveis, esta comparação permite melhor compreender a proposta. A Figura 4.3 descreve a relação entre os níveis do NTCIP e as camadas do modelo OSI, não como um mapeamento direto de funções, mas como referência operacional.

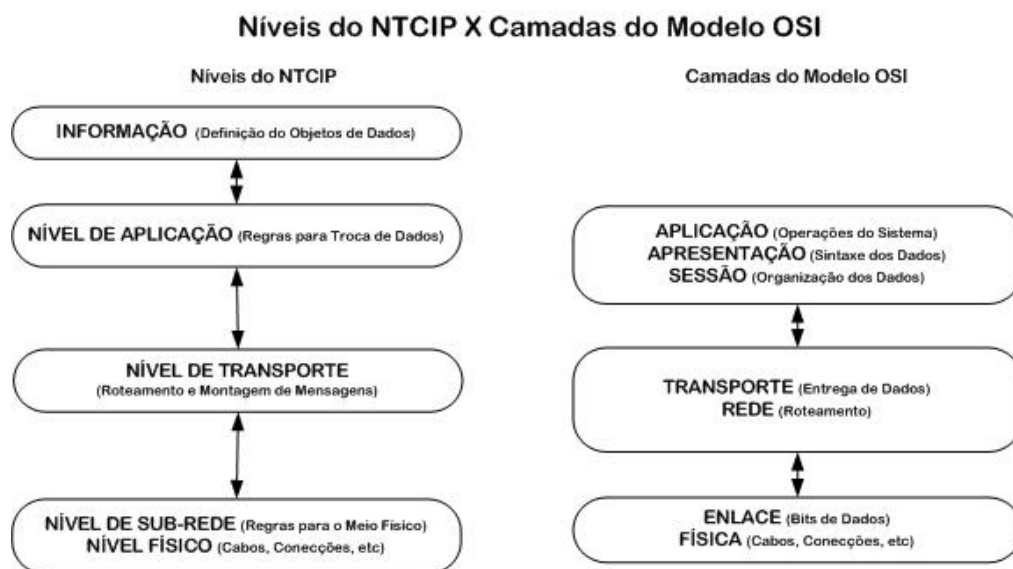


Figura 4.3 – Relação entre níveis e camadas dos modelos NTCIP e OSI/ISO.

4.2.1. Nível de Informação

No nível de informação, são definidos os objetos de dados e seus respectivos perfis de gerenciamento. É aqui que começa surgir a chave para a interoperabilidade e intercambialidade prevista no protocolo, pois são descritos os formatos e localização de cada objeto de dados a serem trocados entre os diferentes dispositivos de um SIT. A tabela 4.1 traz exemplos de alguns dos objetos de dados definidos pela NEMA, para uso na comunicação entre centrais de controle de tráfego e controladores semaforicos. Cada linha da tabela descreve o nome do objeto de dados, seu endereço de referência na árvore MIB, além do tipo e intervalo de valores permitidos.

Tabela 4.1 – Objetos de dados definidos pela NEMA (FHWA, 1997b).

Nome do Objeto	Referência	Nível de Submissão
A.3 – Phase Conformance Group		
MaxPhases	NTCIP 1202.2.2.1	Sempre igual a 16
PhaseNumber	NTCIP 1202.2.2.2.1	Intervalo 1..16
PhaseWalk	NTCIP 1202.2.2.2.2	Inteiro
PhasePedestrianClear	NTCIP 1202.2.2.2.3	Inteiro
PhaseMinimumGreen	NTCIP 1202.2.2.2.4	Intervalo 1..255
PhasePassage	NTCIP 1202.2.2.2.5	Inteiro
phaseMaximum1	NTCIP 1202.2.2.2.6	Intervalo 1..255
phaseMaximum2	NTCIP 1202.2.2.2.7	Intervalo 1..255

Um objeto de dado é definido para atender a demanda de informação gerada por uma central de controle ou por sua instrumentação e representa o estado de uma ou mais variáveis deste dispositivo.

Em uma rede NTCIP, todos os dispositivos devem operar sobre um conjunto comum de objetos, que é denominado dicionário de dados. Qualquer desvio na descrição semântica ou no uso dos objetos, sejam eles obrigatórios ou opcionais, pode causar problemas operacionais ao SIT como um todo. Por este motivo, todos os objetos de dados, sejam públicos ou proprietários, são descritos usando-se a linguagem de notação ASN.1 vista no Anexo 1, com a qual todos definem clara e padronizadamente seu nome, função, status e intervalo de valores possíveis. A Figura 4.4 mostra um objeto de dados da MIB NEMA escrito de acordo com a sintaxe padrão, descrita no Anexo 1.

Exemplo de Objeto de Dados para um Controlador de Sinal de Tráfego Atuado	
Grupo de Parâmetros de Fases - Número Máximo de Fases	
maxPhases	OBJECT-TYPE
	SYNTAX INTEGER (1..255)
	ACCESS read-only
	STATUS mandatory
	DESCRIPTION
	"The Maximum Number of Phases this Actuated Controller Unit supports. This object indicates the maximum rows which shall appear the phaseTable object."
::={phase1}	

Figura 4.4 – Objeto de dados NEMA (FHWA, 1999d).

Observe que a sintaxe e tipos ASN.1 é usada de forma a descrever clara e objetivamente todos os campos necessários à compreensão, codificação e transmissão do objeto de dados. Uma vez que todos os objetos de dados estejam descritos, cabe ao protocolo do nível de aplicação a função de codificação e transmissão destes.

4.2.2. Nível de Aplicação

É no nível de aplicação que os métodos de comunicação começam a ser efetivamente separados por área de atuação, ou seja, a comunicação entre centrais (*Center-to-Center*) ou destas com sua instrumentação (*Center-to-Field*). A justificativa para a subdivisão do nível de aplicação em duas classes torna-se clara a partir da análise de algumas das suas características operacionais.

Como foi dito anteriormente, a comunicação entre centrais dá-se sobre uma rede do tipo ponto-a-ponto, onde centrais estão conectadas fisicamente por cabos de rede em locais pré-definidos. Apesar do maior volume de dados a ser processado e enviados, na comunicação entre as centrais e sua infraestrutura de beira de pista, a frequência e restrições temporais são menores. Assim, para atender a demanda desta subdivisão, são adotados dois padrões para comunicação: o

Common Object Request Broker Architecture – CORBA e o *Data Exchange in ASN.1* – DATEX. A definição de dois padrões de comunicação diferentes foi necessária para que fossem atendidas diferentes demandas da comunicação intersistemas. Além disso, os protocolos podem operar em um mesmo sistema e comunicar-se entre si, desde que existam centrais operando como tradutores entre os dois (FHWA, 1999d).

O CORBA é um padrão difundido dentro da indústria para a comunicação em sistemas distribuídos. Além disso, softwares orientados a objetos podem obter grandes vantagens fazendo uso do CORBA, tanto na implementação quanto na comunicação, já que este é mais eficiente no uso da largura de banda.

Já o DATEX foi projetado para atender a demandas de comunicação a baixo custo e com requisitos de tempo mais restritos. Sua aplicação é recomendada principalmente em sistemas que requerem transmissão de dados em alta velocidade ou que possuam limitação de largura de banda ou ainda no caso de sistemas não orientados a objetos.

Idealmente, ambos os protocolos deveriam estar disponíveis na comunicação entre centrais, o que não impede que apenas um deles venha a ser implementado. Além desses protocolos, o NTCIP também oferece a possibilidade do uso do *File Transportation Protocol* - FTP para troca de arquivos de dados entre centrais. Apesar da comunicação entre centrais ser de fundamental importância para um SIT, os aspectos que mais interessam neste trabalho residem na comunicação entre uma central e sua instrumentação.

Para a comunicação entre as centrais e sua infraestrutura de beira de pista, são utilizados três protocolos padrão: o *Simple Network Management Protocol* – SNMP, o *Simple Transportation Management Protocol* – STMP e o *Simple Fixed Message Protocol* – SFMP, que se encontra em desenvolvimento (FHWA, 2002b). Apesar de cada protocolo ter sido projetado para atender a diferentes demandas de comunicação, todos possuem características operacionais semelhantes às do SNMP, descritas no Anexo 1. Os três protocolos utilizam um conjunto de mensagens get/set, para acessar os dados de um dispositivo SIT.

As diferenças estão no nível de complexidade, tipos de serviços oferecidos e requisitos de implementação. A tabela 4.2 e a Figura 4.5 ilustram as diferenças entre os protocolos de aplicação usados na comunicação com dispositivos.

Tabela 4.2 – Comparação dos protocolos STMP,SNMP e SFMP (FHWA, 1999b).

	SNMP	STMP	SFMP
Pode enviar qualquer elemento de dado?	Sim	Sim	Em desenvolvimento
Uso eficiente da largura de banda?	Ruim	Bom	
Suporta roteamento e conexão discada?	Opcional	Opcional	
Suporte ao conjunto de mensagens SNMP?	Suportado	Limitado a 13	
Facilidade de implementação	Fácil	Difícil	

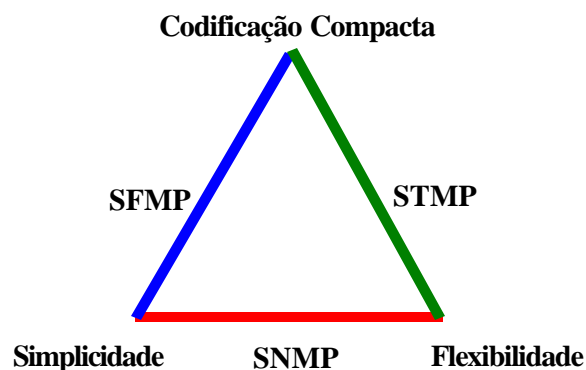


Figura 4.5 – Protocolos de Comunicação *Center-to-Field*

O SNMP é um protocolo de uso geral e foi inicialmente implementado para atender à demanda de gerência de redes Internet. Além de ser muito conhecido e de operação bastante simples, devido ao pequeno número de comandos, adapta-se de forma excelente às necessidades de gerência de dispositivos SIT, dentro da proposta do NTCIP. Entretanto, justamente por sua aplicação genérica, este faz uso de forma bastante ineficiente da largura de banda do canal de comunicação. Isto é, para o envio de um único byte, são necessários aproximadamente 40 bytes de preâmbulo. O NTCIP empenhou-se em propor este protocolo para a indústria de produtos SIT devido à sua ampla utilização na comunidade Internet e fácil implementação.

Resumidamente, as operações do protocolo sobre os dispositivos seguem uma mesma estrutura. Cada estado de um dispositivo é armazenado em uma tabela bidimensional acessível a outro computador, constituindo a *Management Information Base – MIB*. Cada posição da tabela é descrita como um objeto, o qual pode ser acessado ou alterado via SNMP através de seu endereço na árvore global *MB*, como está descrito no Anexo 1. Objetos que estão na tabela podem possuir múltiplas instâncias. Entretanto, objetos de uso exclusivo do dispositivo são denominados de *escalares* e recebem instância (0).

Apesar da implementação do SNMP em dispositivos SIT ser relativamente simples, algumas de suas características operacionais requerem cuidados especiais. Uma delas é permitir que mais de um objeto de dados seja alterado em um único comando *set-request*. Isto implica que diversos objetos sejam atualizados simultaneamente, podendo provocar problemas operacionais no dispositivo e portanto o controle destas atualizações deve ser processado pelo dispositivo receptor.

Já o SFMP pode ser visto como uma simplificação ou uma versão compacta do SNMP (FHWA, 2002b). Em uma análise mais cuidadosa do SNMP, verifica-se que o tamanho e complexidade dos pacotes podem ser reduzidos através de pequenas alterações como:

- Usar um único identificador de conteúdo para os objetos de dados, o qual referenciaria um grupo de elementos de dados;
- Definir uma estrutura de pacote de dados que inclua somente informações requeridas para um determinado tipo de mensagem (por exemplo: um *set-response* não precisa de valores de *echo*);
- Utilizar um conjunto de regras de codificação mais eficiente que o BER, usado pelo SNMP.

Para reduzir efetivamente o tempo de atraso na comunicação, são definidos dois pontos fundamentais na operação do protocolo. O primeiro se refere aos pacotes de dados, que são formados por um único identificador de objeto. Neste tipo de estrutura, um único identificador de grupo é colocado no cabeçalho do quadro e os objetos de dados são separados por um byte de controle, reduzindo o atraso de montagem e o tamanho de cada quadro, em relação ao SNMP. O segundo determina que todos os objetos devem ser localizados sob o nó NEMA da árvore ISO, ou seja, não é permitido o uso de objetos de dados proprietários (FHWA, 2002b).

Outra característica importante do SFMP é o uso de um conjunto de regras de codificação mais eficiente. Enquanto o SNMP utiliza as regras da BER, de acordo com a ASN.1, usando a tupla *Tag-Length-Value* – TLV, ambas descritas no Anexo 1, a proposta do SFMP implementa o método de codificação denominado *Octet Encoding Rules* - OER.

As regras de codificação OER eliminam completamente os campos tipo e tamanho, desde que estes sejam conhecidos. Considerando que o SFMP só utiliza objetos padrão NEMA e que seus identificadores podem ser definidos em um byte, o uso da OER proporciona uma redução significativa do tamanho dos pacotes de dados (FHWA, 2002b). Entretanto, seu uso restringe o acesso de agentes SNMP aos objetos de dados que representam o estado do dispositivo a ser gerenciado. Esta característica também é encontrada em agentes que implementam o protocolo STMP. Para solucionar este problema, o NTCIP define que todos os dispositivos SIT devem implementar uma versão do protocolo SNMP. O Anexo 2 traz uma breve descrição da estrutura de dados e comandos do SFMP.

O protocolo STMP é uma extensão do SNMP e foi implementado pela NEMA para atender a demandas específicas de um SIT, como tempo real ou compensar a baixa velocidade de comunicação dos dispositivos. Conceitualmente o STMP é similar ao SFMP, exceto pelo fato de ter sido projetado para trabalhar com objetos dinâmicos. Um objeto dinâmico é definido no nível de informação e gerado em tempo real no dispositivo. Ao invés de montar pacotes de dados com endereço e valor para cada objeto a ser transmitido, cria-se um endereço de referência, sob o qual coloca-se de 1 a 255 objetos de dados do tipo NEMA, formando um pacote. Este é menor e, portanto, ocupa menos o canal de comunicação, reduzindo a sobrecarga. O uso deste tipo de objeto

permite as estações de gerenciamento definirem de forma flexível suas mensagens, isto é, a estrutura dos blocos de objetos, mas o resultado disso é um aumento significativo da complexidade dos agentes (FHWA, 2002b).

O NTCIP suporta 13 objetos dinâmicos por agente. Teoricamente, a estação de gerenciamento pode configurar cada dispositivo com um conjunto diferente de objetos dinâmicos. Porém, na prática a maioria das estações e dispositivos é configurada com um conjunto comum de objetos dinâmicos. Os objetos dinâmicos são estruturados em forma de tabela, composta por três colunas principais e duas sub-colunas usadas para referenciar efetivamente os objetos de dados da estrutura. A tabela 4.3 descreve a estrutura dos objetos.

Tabela 4.3 – Definição e Configuração de Objetos Dinâmicos (FHWA, 2002b).

dynObjNumber	dynObjConfigOwner	dynObjConfigStatus	DynObj Index	dynObjVariable
1	Objeto dinâmico Proprietário 1	Status do Objeto dinâmico 1	1	<OID 1° dynObj>
			2	<OID 2° dynObj>
		
			255	<OID 255° dynObj>
2	Objeto dinâmico Proprietário 2	Status do Objeto dinâmico 2	1	<OID 1° dynObj>
			2	<OID 2° dynObj>
		
			255	<OID 255° dynObj>
3	Objeto dinâmico Proprietário 3	Status do Objeto dinâmico 3	1	<OID 1° dynObj>
			2	<OID 2° dynObj>
		
			255	<OID 255° dynObj>
...
13	Objeto dinâmico Proprietário 13	Status do Objeto dinâmico 13	1	<OID 1° dynObj>
			2	<OID 2° dynObj>
		
			255	<OID 255° dynObj>

Para compreender melhor a idéia de objeto dinâmico, quatro conceitos devem ser reconhecidos. São eles: Elemento de Dados, Objeto Bloco, Quadro de Dados e Pacote de Dados.

Um elemento de dados é uma representação formal de uma unidade de informação de interesse e indivisível, como por exemplo, uma das instâncias de um objeto MIB. Já um objeto bloco é um objeto com uma sintaxe *OERstring*, ou seja, uma palavra que contém um elemento de dado que já tenha sido serializado de acordo com as regras da *Octet Encoding Rules* - OER.

Para que se gere um quadro de dados, um ou mais objeto bloco devem ser agrupados sob um mesmo identificador. Então, quando este quadro de dados é serializado juntamente com seu identificador formando um *byte stream*, é criado o pacote de dados.

Assim, quando um pacote de dados é gerado em tempo de execução (dinamicamente), com um identificador curto e codificado dentro das regras da OER, este é chamado Objeto Dinâmico. O Anexo 3 traz um exemplo de objeto dinâmico. É importante salientar que objetos dinâmicos são acessados somente pelo STMP e conseqüentemente não são objetos verdadeiros para o SNMP. A Figura 4.6 ilustra o processamento para identificar qual foi o protocolo usado na codificação de uma mensagem recebida por um dispositivo SIT.

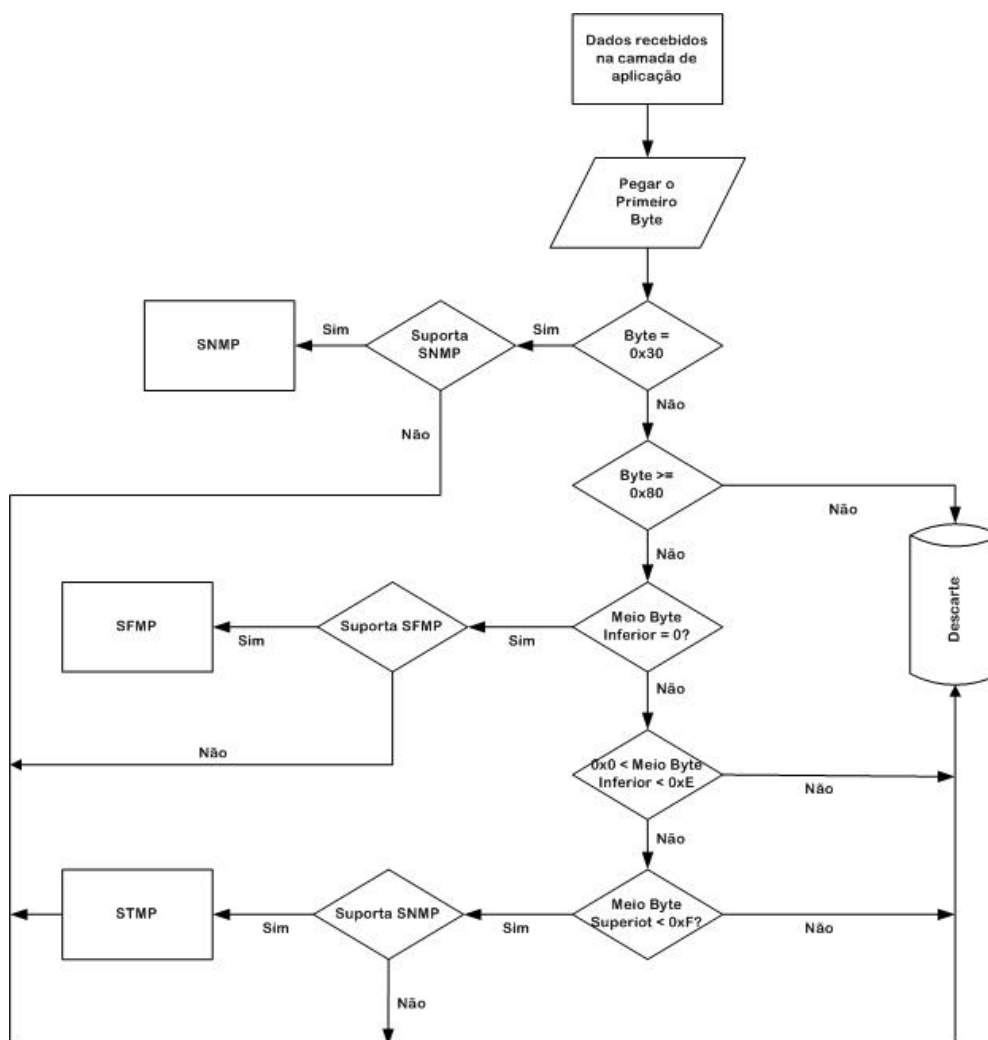


Figura 4.6 – Processo para determinar o protocolo a ser utilizado (FHWA, 2002b).

Observe que a definição de qual protocolo será utilizado pelo receptor para decodificar o pacote de dados recebido depende da interpretação do primeiro byte do quadro. Isto não é uma opção feita pelos projetistas do NTCIP, mas se deve às características do SNMP e agregadas ao SFMP e STMP, que coloca no primeiro byte de suas mensagens um rótulo que identifica o tipo de seqüência de dados utilizado, como está descrito no Anexo 1.

Pode-se dizer que os protocolos para gerenciamento do transporte, especificados no NTCIP em seu serviço de aplicação, procuram prover 100% de interoperabilidade com o padrão Internet SNMP. Além disso, a proposta estende este protocolo através do STMP e SFMP, procurando atender as necessidades de flexibilidade, simplicidade e tamanho mínimo de pacote de dados, dos ambientes de transporte. Entretanto, na maioria dos casos, estes três requisitos são conflitantes (FHWA, 2002b).

4.2.3. Nível de Transporte, Sub-Rede e Físico

Estes níveis juntos são responsáveis pela montagem/desmontagem, roteamento, entrega e conexão física dos dispositivos do SIT. O conjunto de tecnologias e padrões de hardware descritos na proposta do NTCIP para prover estes serviços é aquele que além de atender à demanda de comunicação, seja o mais conhecido e utilizado pela indústria de produtos para SIT (FHWA, 1999d). Por se tratar de uma proposta aberta, a pretensão dos projetistas do protocolo é que, com o desenvolvimento de novas tecnologias, todas as demandas de comunicação dos SIT possam vir a ser atendidas pelo NTCIP. No entanto, nem todas as combinações são possíveis e, como foi dito anteriormente, as escolhas nos níveis superiores têm influência sobre as que serão feitas neste. Assim, o uso de um padrão no nível de transporte pode impedir ou determinar a utilização de outro no nível de sub-rede ou físico.

O NTCIP permite que um conjunto significativo de aplicações SIT possa ser implementado, utilizando seus métodos de comunicação. Porém, alguns subsistemas definidos na *National ITS Architecture* não podem ainda ser atendidos pela versão atual do protocolo.

4.3. Aspectos da Comunicação entre Centrais (*Center-to-Center*) e com seus Dispositivos.

O objetivo principal do NTCIP na comunicação entre as centrais e sua infraestrutura junto à pista é de integrar sistemas e dispositivos SIT utilizando um único canal de comunicação que possa ser compartilhado por todos. Por exemplo, com a colocação de um software apropriado em um computador, painéis de mensagens dinâmicas e controladores semafóricos podem compartilhar o mesmo canal de comunicação (FHWA, 1999d). Esta característica reduz

significativamente o custo de implantação e manutenção de um SIT, mas só é possível se a proposta de padronização dos métodos de comunicação do NTCIP for utilizada por todos os dispositivos do sistema.

No outro extremo da proposta, na comunicação entre centrais, que usualmente se dá entre computadores dispostos em locais fixos e sobre uma LAN ou WAN, a idéia é que sejam usadas as redes públicas de dados. O que motiva os planejadores do NTCIP a definir uma infraestrutura de comunicação para a integração de sistemas com estas características, é o fato de que um dos componentes mais caros na implantação de um SIT é a rede de comunicação (FHWA, 1999d). A Figura 4.7 ilustra um exemplo de integração de sistemas e dispositivos para SIT usando NTCIP.

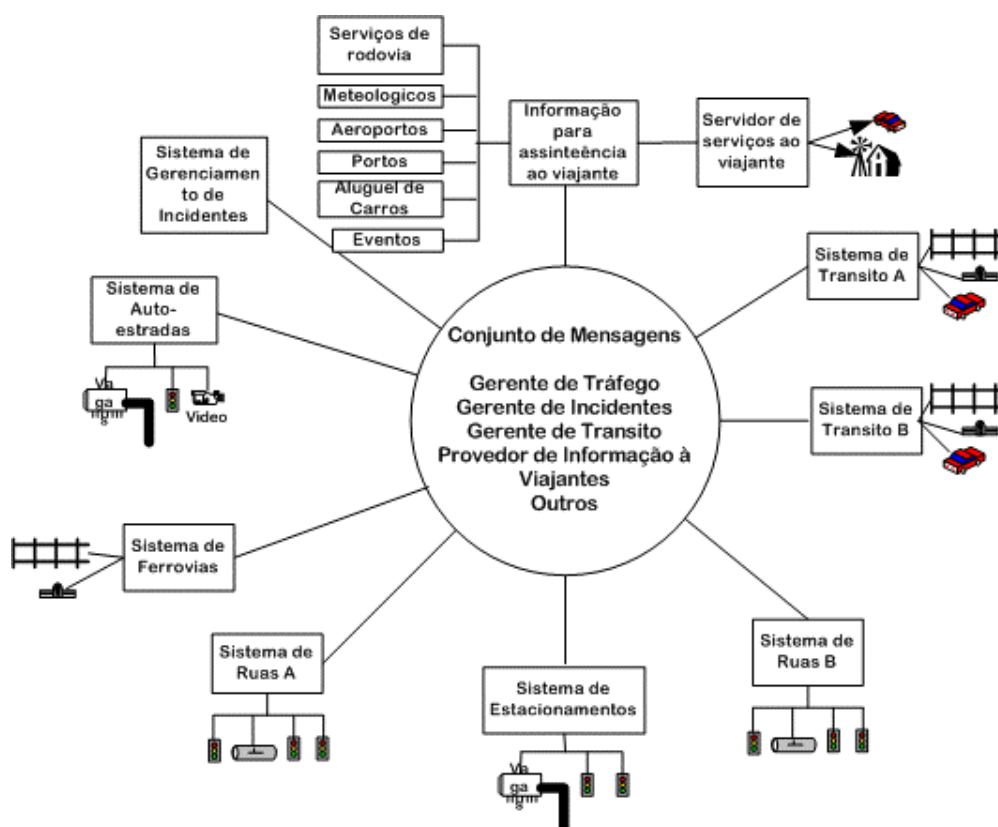


Figura 4.7 – Exemplo de integração de um SIT usando NTCIP.

O NTCIP ainda não atende todo tipo de sistema ou dispositivo. Existe um conjunto de aplicações que por possuírem características especiais, seja de tempo ou tecnológicas, não são ainda suportados pelo protocolo. Algumas dessas aplicações são:

- Leitura ou escrita de rótulos em veículos em movimento
- Vídeo em *full motion*

- Transmissão de dados entre veículos
- Comunicação para controle veicular

Espera-se que, com o desenvolvimento de novos padrões, estas e outras demandas possam vir a ser atendidas dentro do NTCIP. Porém, a característica mais restritiva à interoperabilidade nos SIT atuais é que grande parte dos dispositivos instalados não suporta parcial ou integralmente o NTCIP e nem permite uma atualização para este.

Muitos dispositivos legados operam sobre padrões e tecnologias proprietárias impedindo o uso de um canal de comunicação compartilhado. Para casos em que ambas as tecnologias venham a ser implementadas conjuntamente, podem ser mescladas diferentes infraestruturas de comunicação, permitindo assim, manter dispositivos legados operando conjuntamente com padrão NTCIP. O uso de canais separados exige que tradutores gerenciados por um único computador gerente sirvam de ponte entre os dispositivos legados e aqueles em concordância com o padrão NTCIP. A Figura 4.8 ilustra um exemplo de migração de um sistema não-NTCIP para NTCIP.

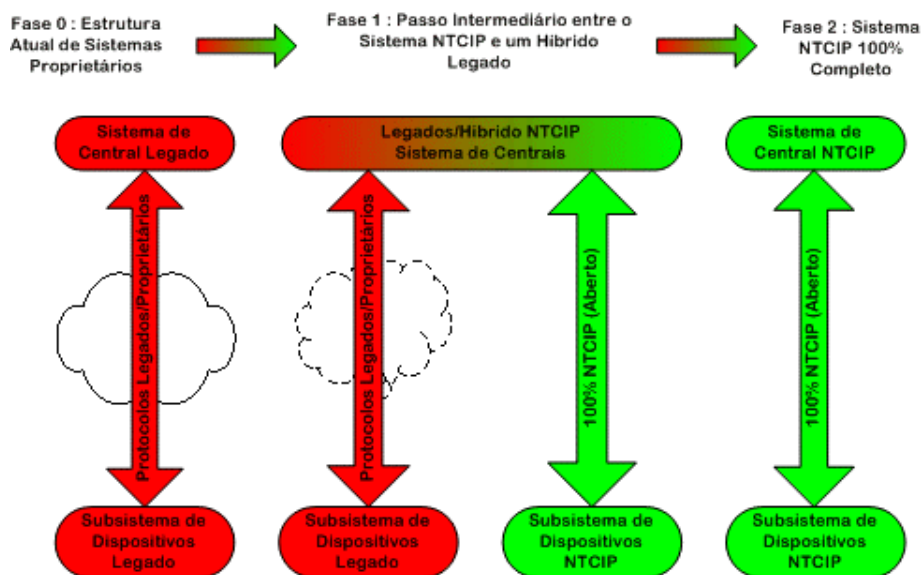


Figura 4.8– Migração de um sistema legado /proprietário para um NTCIP puro.

4.4. Considerações Finais

Neste capítulo estão descritos os princípios fundamentais da infraestrutura de comunicação do NTCIP. Evidentemente, não foi possível fazer com que as descrições de cada um dos níveis do protocolo fossem completas. Isto se deve principalmente ao grande número de

tecnologias, padrões e conceitos envolvidos na compreensão de cada nível do protocolo e das interações entre estes.

Seja por questões técnicas, operacionais ou para o favorecimento da compreensão do protocolo, este foi dividido em dois grandes grupos de comunicação: um que trata da troca de dados entre centrais de gerenciamento e outro para a comunicação entre centrais e sua instrumentação junto à pista. Em ambos os casos, o NTCIP oferece um conjunto de opções tecnológicas distribuídas em níveis, permitindo a combinação destas tecnologias de forma a montar uma pilha de protocolos. Em cada um dos níveis da proposta, estão descritas as abrangências e limitações impostas na escolha de cada uma delas.

Porém, o aspecto mais importante do protocolo não advém do arcabouço tecnológico, mas da proposta de integração e conectividade. O objetivo central do NTCIP é oferecer aos desenvolvedores um conjunto de padrões e tecnologias abertas, que forneçam condições para a intercambialidade e interoperabilidade entre sistemas e dispositivos SIT. Em sua proposta de conectividade, o NTCIP não restringe o uso de outros protocolos; ao contrário, procura integrar-se aos sistemas existentes, até que estes sejam definitivamente substituídos por ele.

Apesar de conceitualmente simples, existem inúmeros detalhes do NTCIP não descritos até agora, mas fundamentais à sua implementação. O capítulo 6 descreve os passos para a construção de uma aplicação de comunicação usando o NTCIP e detalha melhor uso de padrões e de tecnologias na implementação do protocolo.

5. ESTUDO DE CASO PARA A IMPLEMENTAÇÃO DO NTCIP

5.1. Introdução

Neste capítulo, estão detalhados os passos para a construção de um aplicativo NTCIP dentro da proposta da *National ITS Architecture*-NIA para a comunicação entre centrais e sua infraestrutura de beira de pista entre dispositivos SIT. Inicialmente está descrito o escopo da aplicação, suas restrições e atributos desejados, dentro da perspectiva de oferecer ao leitor uma ferramenta suficientemente clara, a fim de servir como um roteiro para futuras aplicações.

Como se pode observar claramente nos capítulos anteriores, conceitualmente o processo de comunicação do NTCIP, tanto entre centrais quanto destas com sua infraestrutura junto à pista, faz bastante uso de protocolos conhecidos. Entretanto, uma aplicação real requer um nível de refinamento elevado dos conceitos anteriormente expostos, principalmente dos que se referem à estrutura de dados e codificação, descritos no Anexo 1. Assim, para não estender demasiadamente cada seção, detalhes sobre o armazenamento, codificação e decodificação e transmissão de dados são apenas referenciados no decorrer deste capítulo.

O escopo da aplicação é desenvolver um conjunto de processos de software que, a partir de um canal físico de comunicação entre um computador, denominado central de controle e um dispositivo SIT virtual se estabeleça uma conexão utilizando UDP/IP. Esta conexão deve fornecer condições para que a central gerencie um ou mais dispositivos SIT, sobre o mesmo canal, utilizando-se do protocolo padrão NTCIP.

5.2. Descrição da Aplicação

Apesar do NTCIP oferecer uma solução de comunicação entre a maioria dos módulos e subsistemas da NIA, (vide Figura 4.1) a aplicação desenvolvida aqui se restringe a implementar a troca de dados e comandos entre uma central de controle de tráfego urbano e sua instrumentação (sensores e atuadores). Após definir-se os atributos principais da aplicação, tais como os dispositivos que estarão envolvidos, pode-se então delimitar que padrões serão usados em cada nível do NTCIP.

Existem diversos fatores a serem observados para a escolha das tecnologias e padrões que são utilizados na construção de um sistema de comunicação para SIT. Na proposta do NTCIP, condições ambientais, custos operacionais, velocidade desejada, tipo de dispositivos SIT utilizados, além de outros fatores, devem ser considerados no momento da escolha das tecnologias de comunicação. Porém, por tratar-se de uma aplicação a ser executada sobre um ambiente virtual,

foram considerados dois aspectos como os mais relevantes: o primeiro é que, por tratar-se de um trabalho de reconhecimento e avaliação das tecnologias e características funcionais do NTCIP, que possam ser aplicadas na comunicação entre dispositivos de SIT no Brasil, devia-se escolher um perfil de pilha de protocolos aplicável à indústria nacional. O segundo refere-se ao próprio ambiente acadêmico no qual será executada a simulação que, por si só, restringe o leque de padrões de hardware e software disponíveis à aplicação. O conjunto formado pelo simulador SITRA-B+ e o aplicativo desenvolvido simulam virtualmente a comunicação entre um conjunto de dispositivos de campo (instalados junto à via) e uma central de gerenciamento de tráfego (colocada em um local remoto, como ilustra a Figura 5.1).

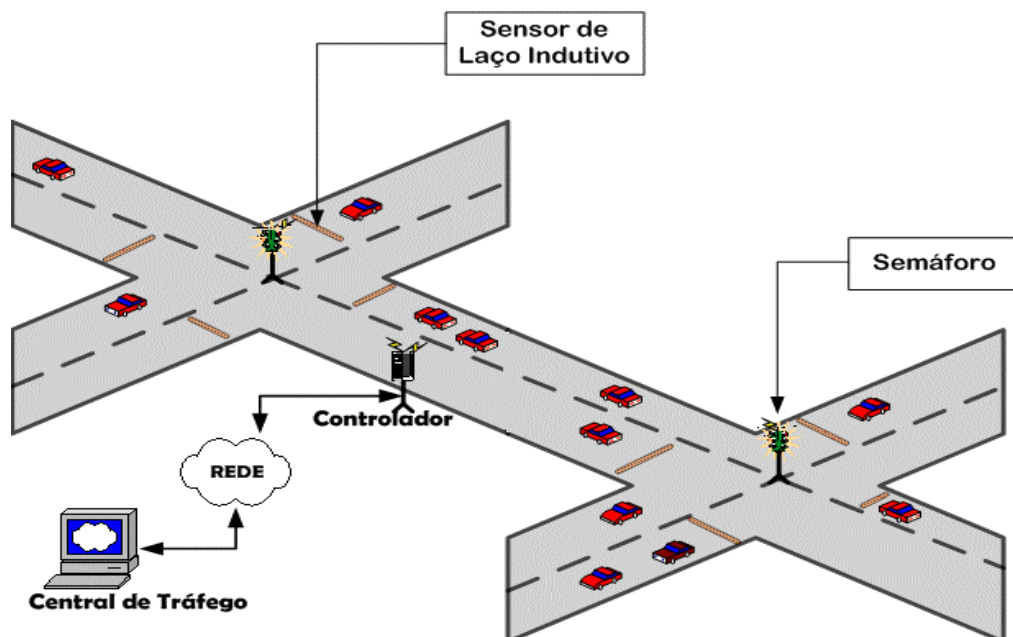


Figura 5.1 – Ilustração dos dispositivos envolvidos na aplicação NTCIP.

Para fins deste trabalho, consideram-se os aspectos até o nível de aplicação, sendo usadas rotinas disponíveis para comunicação UDP/IP (Borland, 2001). A simulação não requer escrita de software ou suporte de hardware especial para usar *Ethernet*/par trançado. A Figura 5.2 ilustra a escolha de padrões e tecnologias feita em cada um dos níveis do NTCIP, definindo uma pilha de protocolos a ser utilizada na confecção da aplicação de software.

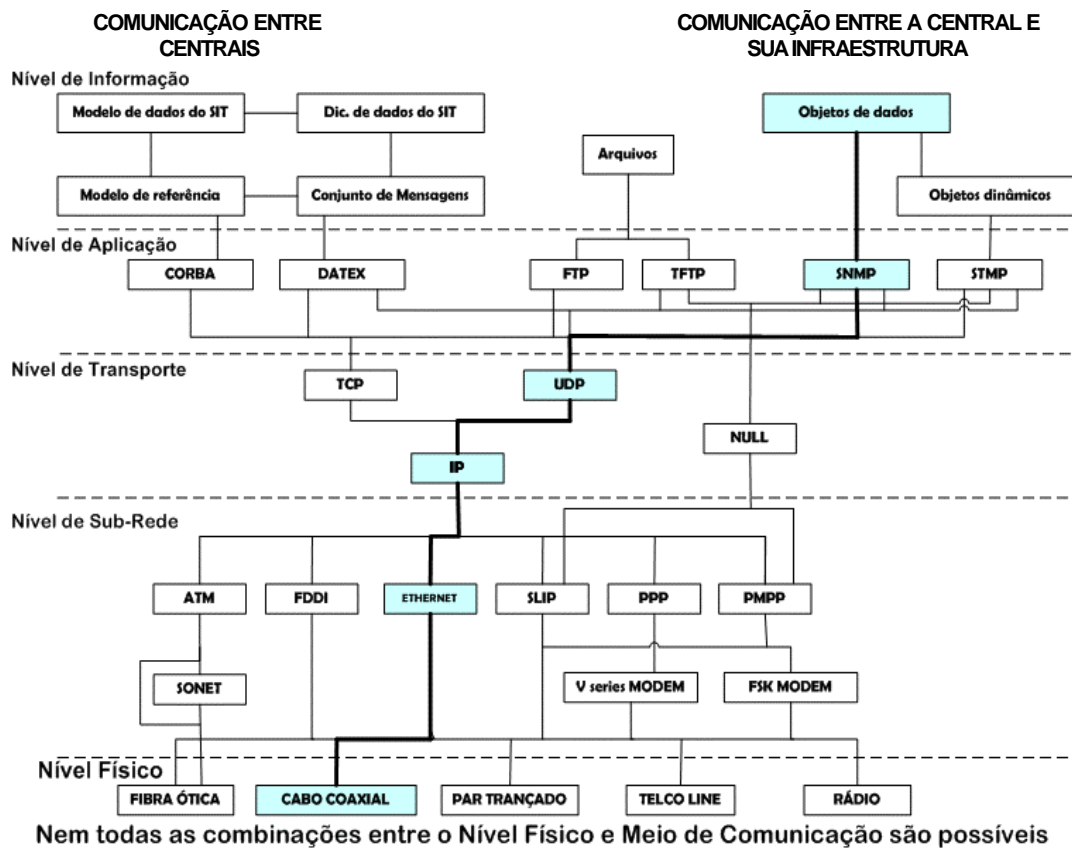


Figura 5.2 – Pilha de protocolos NTCIP usados na Aplicação (FHWA, 1999b).

Uma vez reconhecidos os aspectos relativos ao ambiente de aplicação, tanto os que se referem ao hardware quanto aos de software e suas conseqüentes restrições operacionais, pode-se então realizar o passo seguinte que é, a definição da estrutura do sistema. A fim de manter a coerência entre o referencial bibliográfico e este trabalho, os critérios de análise, implementação e validação desta aplicação, baseia-se na análise estruturada. Apesar não existirem restrições ou sugestões quanto aos processos de análise, projeto e desenvolvimento de produtos baseados no padrão NTCIP, a NEMA utiliza as ferramentas da análise estruturada, descrevendo seus processos e fluxos de dados através de Diagramas de Fluxo de Dados – DFD.

5.3. Arquitetura da Aplicação

Através da implementação de uma central de controle de tráfego e de um controlador semafórico virtual, busca-se compreender e avaliar as implicações tecnológicas do uso do NTCIP na comunicação entre dispositivos SIT.

O modelo de software proposto é formado por 4 (quatro) entidades básicas: o simulador

de tráfego SITRA-B+ (SODIT, 2002), uma estratégia de controle de tráfego e dois processos SNMP, um agente, que representa o controlador semafórico e outro o gerente, que atua como central de controle de tráfego.

Os módulos gerente e agente, são as entidades principais do aplicativo. São eles que efetivamente fornecem as condições para a troca de dados na forma proposta pelo protocolo NTCIP. Estes módulos operam respectivamente, no computador da central de controle de tráfego e no controlador semafórico, instalado junto à pista. Nesta aplicação, os sinais gerados pelo controlador, normalmente captados da instrumentação, são substituídos por dados gerados pelo simulador de tráfego. Em princípio, os controladores simulados podem ser substituídos diretamente por dispositivos reais, bastando para isso que estes últimos usem o protocolo padrão, ou seja, sejam *intercambiáveis*.

A Figura 5.3 descreve o DFD da aplicação, dentro do modelo de comunicação entre dispositivos, proposto pelo NTCIP e desenvolvido neste trabalho.

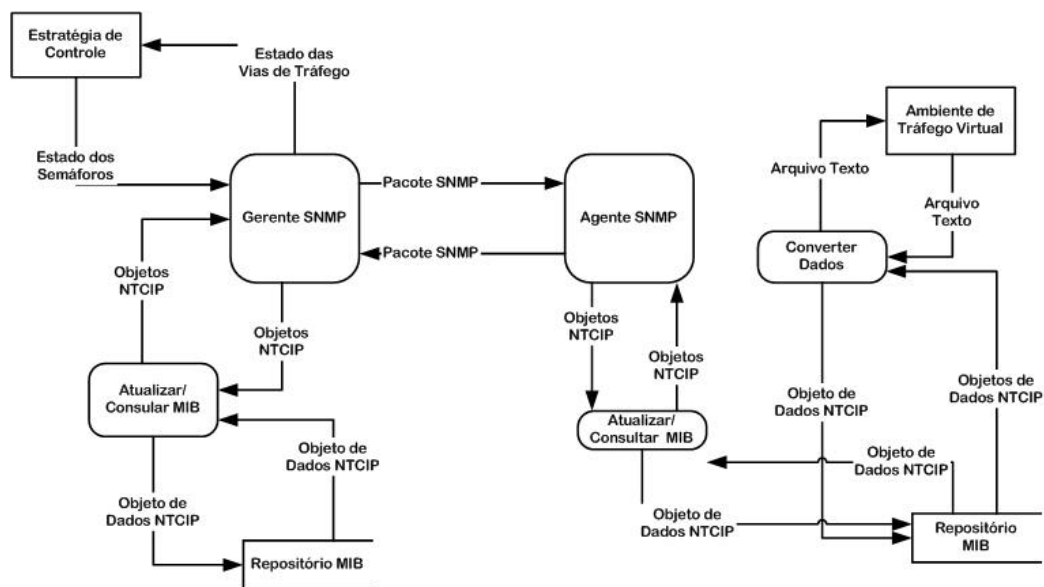


Figura 5.3 – Diagrama de Fluxo de Dados da Aplicação.

A proposta do NTCIP não sugere o uso de orientação a objetos ou qualquer outro conceito ligado à implementação. Todavia, as referências bibliográficas utilizam-se do conceito de Diagrama de Fluxo de Dados - DFD para definir os módulos e a comunicação entre os processos do sistema. Assim, visando manter a coerência entre o referencial bibliográfico do NTCIP e este documento, também se fará uso de DFDs para descrever a operação do aplicativo.

Em princípio, a implementação do NTCIP é simples. A aplicação consiste apenas de um processo (gerente), que envia repetidamente mensagens a outro processo (agente), com as operações específicas que este deve executar. A maioria das mensagens importantes é formada por

uma seqüência de *get* e *set*, juntamente com objetos de dados ASN.1. O agente recebe as mensagens, as decodifica e executa a operação. Além disso, o agente é responsável pela obtenção dos dados, junto ao simulador e posterior atualização de sua MIB.

Na prática muitos detalhes de implementação acabam por complicar o código. Tratam-se de questões como, codificação e decodificação dos tipos ASN.1 ou da montagem de *stream* de dados e outros. Para simplificar e tornar mais claro ao leitor quais as trocas de mensagens executadas entre os processos da aplicação, os requisitos e operações estão descritos por nível, da mesma forma que o protocolo foi apresentado. Em cada um dos níveis, informação, aplicação, transporte, sub-rede e físico, são descritas operações específicas, estrutura de dados ou hardware necessários, bem como sua relação com as camadas superiores e inferiores da pilha de protocolos.

5.3.1. Nível de Informação

Neste nível são descritos os objetos de dados que serão usados para a comunicação entre a central de controle de tráfego e o controlador semafórico. A NEMA define um amplo conjunto de objetos de dados que se referem a este tipo de operação. Entretanto, devido às peculiaridades da aplicação, que não opera sobre um controlador físico, mas em um ambiente simulado, alguns dos objetos utilizados na comunicação são de uso exclusivo ou como define a NEMA, proprietários. O uso de objetos de dados proprietários é permitido, como já foi dito no Capítulo 4, desde que estes objetos sejam descritos e codificados dentro das regras definidas pela ASN.1. Assim, para atender às demandas do próprio simulador e dos agentes SNMP foram criados três objetos de dados sob um nó fictício na árvore MIB, como está descrito abaixo.

```
simulador OBJECT IDENTIFIER ::= { iso(1) org(3) dod(6) internet(1)
    private(4) enterprises(1) simulador(5000) }
```

FaseVerde OBJECT-TYPE

SYNTAX INTEGER (1..255)

ACCESS read-only

STATUS mandatory

DESCRIPTION "Representa o número da fase de verde que será ativado pelo simulador SITRA-B+. Este objeto indica o número máximo de fases suportado pelo sistema"

```
::= { simulador }
```

filaNaVia **OBJECT-TYPE**

SYNTAX VisibleString

ACCESS read-only

STATUS mandatory

DESCRIPTION “Representa a quantidade de veículos sobre a pista no momento da leitura do controlador. Este objeto indica o número de veículos parados ou em movimento dentro de um arco a ser definido nos parâmetros do simulador SITRA-B+ “.

::= {simulador}

A estrutura fictícia da árvore MIB criada exclusivamente para esta aplicação, segue o padrão definido pela MIB-II [RFC1155] e utilizada pela NEMA, como ilustra a Figura 5.4.

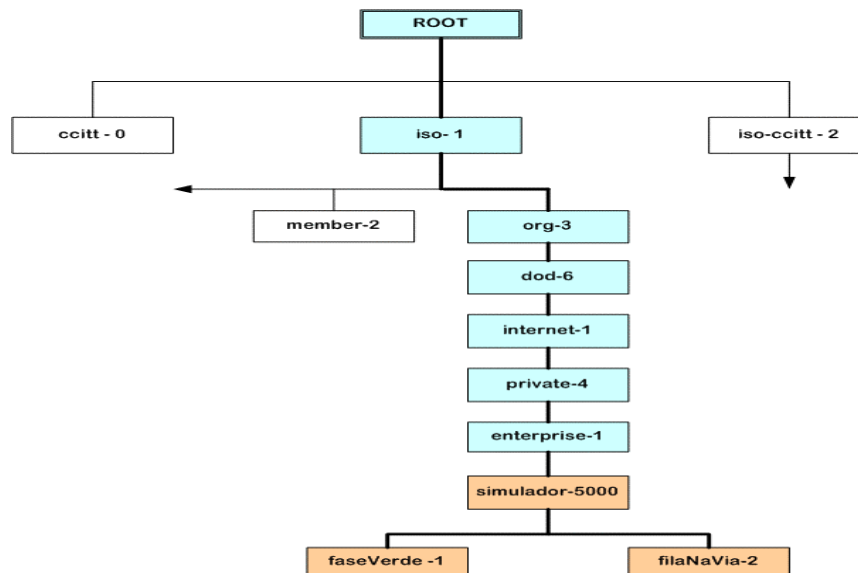


Figura 5.4 – Estrutura da árvore MIB para a aplicação.

Seguindo a norma de construção de endereço para objetos MIB, o identificador de objeto (simulador-5000) e os objetos de dados (faseVerde-1 e filaNaVia-2) possuirão os seguintes endereços OID:

- (OID) – Simulador-5000: 1.3.6.1.4.1.5000
- (OID) - faseVerde-1: 1.3.6.1.4.1.5000.1
- (OID) - filaNaVia-2: 1.3.6.1.4.1.5000.2

Uma vez que os objetos de dados e seus respectivos identificadores estejam claramente definidos e estruturados dentro do nível de informação, o passo seguinte é estabelecer quais os protocolos a serem usados no nível de aplicação.

5.3.2. Nível de Aplicação

No nível de aplicação, estão definidos os protocolos possíveis para a comunicação entre as centrais (*Center-to-Center*) ou destas com sua infraestrutura junto à pista (*Center-to-Field*). O aplicativo desenvolvido neste trabalho visa atender à demanda de comunicação entre uma central de tráfego e seus controladores semafóricos (*Center-to-Field Communications*). Isto restringe a escolha a três protocolos neste nível, o SNMP, STMP ou SFMP, descritos no capítulo 4. O uso do SNMP como protocolo de aplicação nesta implementação, se deve a três motivos principais:

- O protocolo atende plenamente às demandas de comunicação da aplicação;
- Facilidades de implementação do código;
- Farta documentação disponível.

Além dos motivos acima, o SNMP é um protocolo extensivamente utilizado por administradores de rede e outros profissionais desta área, tornando-se, assim, uma ferramenta facilmente assimilável pelos desenvolvedores de produtos SIT.

Quanto às suas características operacionais, o principal problema deste protocolo é operar sobre UDP/IP, o que significa a inexistência de garantia de entrega das mensagens. A tarefa de verificação dos pacotes de dados é de responsabilidade dos agentes e deve ser escrita pelo desenvolvedor. Além disso, a aplicação gerente deve garantir os controles para que as mensagens sejam recebidas e dentro dos tempos de operação estabelecidos no sistema.

Nas aplicações que utilizam o protocolo NTCIP, o software gerente serve também de interface para outras aplicações, como: estratégia de controle, serviços de informação, dispositivos de controle e o operador humano, entre outras. A Figura 5.5 ilustra a operação de um processo SNMP.

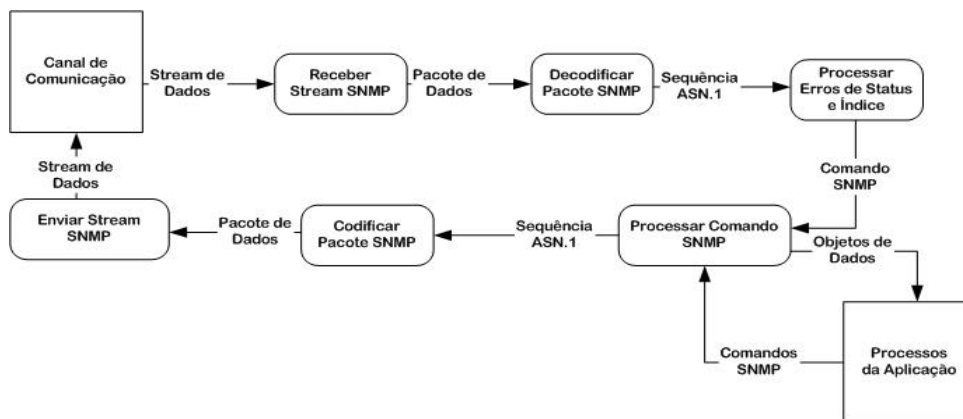


Figura 5.5 – DFD do processo de Comunicação de um agente SNMP.

O processo acima descrito ocorre no nível de aplicação do NTCIP, onde as questões ligadas à comunicação física dos dados são abstraídas, já que o controle de transmissão e recepção destes é de responsabilidade dos níveis inferiores. Em uma aplicação real, os níveis de transporte, sub-rede e físico do NTCIP devem ser cuidadosamente tratados pelo desenvolvedor e, em alguns casos, implementados.

5.3.3. Nível de Transporte, Sub-rede e Físico

São as características do hardware dos dispositivos SIT e o tipo de protocolo do nível anterior que definem quais os padrões possíveis à aplicação nestes três níveis.

No nível de transporte, o número de opções é limitado a dois: ou se usa o pacote UDP sobre IP ou NULL. Além disso, o hardware do nível de sub-rede também cria restrições, por exemplo, dispositivos SIT que fazem uso de tecnologia *Ethernet* não podem operar sob a opção de protocolo NULL, como podemos verificar na Figura 5.2.

Quanto ao nível físico, não existem restrições dos níveis anteriores à escolha de qualquer tipo de tecnologia. Porém, permanecem as restrições impostas pelo hardware e, além disso, cada tipo de meio físico impõe uma largura de banda máxima, que serve como parâmetro de escolha, dependendo das restrições temporais de cada aplicação.

5.4. Arquitetura Interna

Esta seção descreve o funcionamento interno dos processos gerente e agente, desenvolvidos em C++. Estes dois softwares compartilham classes, objetos e um conjunto de regras para codificação, transmissão e decodificação de dados. Porém, têm características operacionais distintas.

Para a construção destes softwares foi necessário um estudo minucioso das regras de codificação BER, da arquitetura do SNMP versão 1, das RFCs ligadas a MIB, além dos padrões de comunicação Internet envolvidos. Assim, foi possível a construção de um PDU padrão NTCIP, que pôde ser codificado, transmitido e decodificado dentro da proposta de comunicação do protocolo.

5.4.1. Implementação da Classe MIB

O objetivo desta classe é armazenar o conjunto de variáveis de estado do dispositivo, permitindo que estas sejam consultadas ou alteradas por comandos *get/set* do SNMP. Observe que esta classe apenas armazena os estados. Isto se deve a diferenças operacionais do gerente e do agente. Enquanto no processo agente esta estrutura de dados deve ser atualizada constantemente,

ou seja, toda vez que o estado do dispositivo mudar, no processo gerente estas alterações dependem de outros fatores, como, algoritmo de controle, operações externas, etc.

O SNMP pode armazenar informações sobre cada variável MIB. Na implementação deste protocolo de aplicação, estas variáveis podem ser descritas usando um vetor, em que cada posição corresponde a um objeto MIB. O código descrito no quadro 5.1 define a estrutura MIB e a inicialização do vetor, utilizando os objetos de dados criados para esta aplicação.

Quadro 5.1 – Código da estrutura de dados de uma MIB.

```

/* mib.h  estrutura de dados para a implementação de uma MIB */

struct mib
{
    char *nome;
    char *prefixo;
    char *oid;                /* endereço OID */
    int *tipo_de_dado;        /* tipo ASN.1 */
    Bool tipo_de_objeto;     /* write */
    struct mib *proximo_objeto; /* apontador para o
                                próximo objeto */
};

struct mib MIB[ ];
int max_objetos;
/* mib.c inicialização do texto */
#include <mib.h>
#define v_Agregado 06
#define v_Integer 02
#define v_VisibleString 1A

struct mib MIB[ ]= {
    {"simulador", " ", {{5000}, 1}, v_Agregado, FALSE, (char *)
    &faseVerde },
    {"faseVerde", " ", {{1}, 0}, v_Integer, FALSE, (char *) &
    filaNaVia},
    {"simulador", " ", {{2}, 0}, v_VisibleString, FALSE, (char *)
    &filaNaVia}}

```

Observe que o quadro anterior descreve uma estrutura dinâmica para o armazenamento dos objetos. Isto não é pré-requisito para o funcionamento dos agentes SNMP. Porém, o uso de outras estruturas estáticas tende a aumentar a complexidade dos agentes no acesso e armazenamento dos objetos de dados. Outro fato relevante na aplicação é que não existem restrições ao tipo de linguagem a ser usado para a implementação da MIB. Esta pode ser construída em qualquer linguagem suportada pelo controlador ou central de tráfego, ou mesmo implementada em hardware, dependendo da conveniência, custo, suporte ou necessidade de performance do sistema.

5.4.2. Implementação da Classe Codificar

A classe codificar é responsável pela montagem e desmontagem da PDUs. É ela que manipula as informações trocadas entre os processos gerente e agente.

Para facilitar a compreensão desta classe, seu funcionamento pode ser dividido em cinco tarefas, que são executadas em ambos os processos:

1. Recepção do PDU: o pacote é recebido através dos processos de comunicação e disponibilizado em memória RAM para decodificação.
2. Leitura da *stream* de dados: é verificado qual é o tipo de protocolo que está sendo usado e, após isto, quais são os comandos SNMP estão sendo solicitados ou se o pacote traz um TRAP. O tratamento dos TRAPs é relativamente complexo e é recomendada uma avaliação mais profunda da bibliografia, antes de sua implementação.
3. Decodificação da estrutura TLV: uma vez identificado o comando, a PDU deve ser decodificado seguindo a regra BER sobre uma estrutura TLV, como descrito no Anexo 1 deste trabalho. Assim, o primeiro campo traz o tipo de seqüência de dados; o segundo, o número de bytes que deve ser lido, e o terceiro, o valor de dado a ser interpretado. É importante lembrar que, em um pacote SNMP, o dado propriamente dito (objeto de dado), está na última TLV; as informações anteriores fazem parte do cabeçalho da mensagem.
4. Verificação de Erros do SNMP: O SNMP possui dois campos de Erro. Estes campos devem possuir zero em seu valor de dado. Caso não seja este o valor recebido, deve-se proceder conforme as recomendações do SNMP em sua tabela de erros RFC1157 (IAB, 1993).
5. Consulta ou Atualização da MIB: uma vez que a PDU tenha sido decodificada e validada, então, deve-se atualizar o repositório MIB na memória RAM.

O Anexo 5 traz parte do código usado para a decodificação de uma mensagem SNMP utilizada nos processos gerente e agente.

Apesar das regras de codificação e de decodificação serem as mesmas, tanto para transmissores quanto para receptores de dados, a manipulação do pacote é diferente. Enquanto, na

recepção, o pacote de dados é interpretado do início para o final, na transmissão a montagem se dá na ordem inversa, ou seja, primeiramente codifica-se o dado e em seguida agregam-se as informações de tamanho e *tag*. Isto ocorre devido às características do SNMP, que utiliza a estrutura *Tag-Length-Value* - TLV, para firmar seu pacote de dados. Assim, na transmissão, o emissor deve primeiramente codificar o valor do dado, descobrindo seu tamanho, e agregar o respectivo rótulo da operação, até completar todos os passos da construção do pacote de dados. Já, na recepção, ocorre o contrário. O receptor desmonta a mensagem de dados a partir do início, verificando se todas as partes estão corretas, até encontrar, na última posição, o valor do dado.

A Figura 5.6 descreve um pacote de dados e seu cabeçalho SNMP, gerado por uma aplicação gerente.

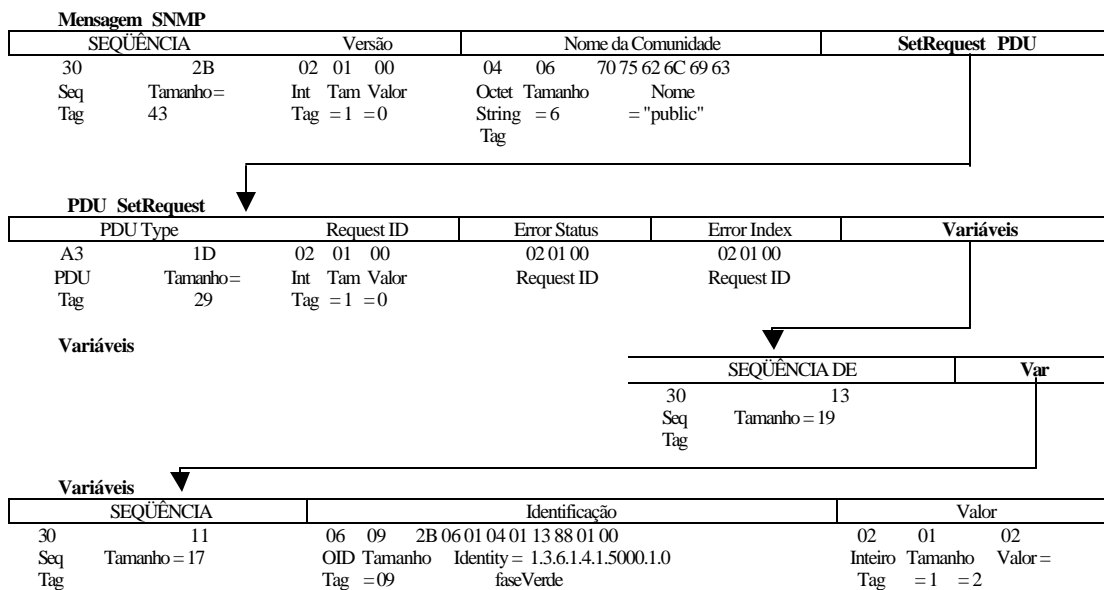


Figura 5.6 – Estrutura de uma mensagem SNMP (FHWA, 1999d).

5.4.3. Implementação da Classe Comunicar

Esta classe tem a função de prover, aos processos gerente e agente, o suporte para a transmissão e recepção de mensagens via um canal UDP/IP. Além da função de comunicação, a classe comunicar deve executar a verificação da recepção dos pacotes, já que o UDP não garante a entrega dos dados.

A aplicação desenvolvida neste trabalho utiliza-se dos protocolos UDP/IP sobre tecnologia *Ethernet* e uma mistura de meios físicos de comunicação. A escolha dos padrões utilizados abaixo do nível de aplicação se deve a dois fatores primordiais: as restrições operacionais do simulador SITRA-B+ e as disponibilidades tecnológicas do ambiente acadêmico. Desta forma,

conceitos como *Ethernet*, par trançado e outros, vêm encapsulados no conceito de Intranet, o qual é utilizado como suporte para a comunicação entre a central de controle de tráfego e sua instrumentação virtual.

5.5. Interface do Usuário

Neste item são apresentados os detalhes referentes ao funcionamento externo dos processos agente e gerente, suas janelas, botões e linhas de comando. Por se tratar de uma simulação, ambos os processos, quando são executados, permanecem parados até que o simulador SITRA-B+ inicie. Uma vez que o simulador de tráfego começa a atualizar seus arquivos de saída, o processo agente notifica o gerente que inicia seu controle sobre o primeiro, usando o protocolo NTCIP. A partir deste ponto, todas as alterações no ambiente de simulação são feitas remotamente, sob o critério de uma estratégia de controle implementada no processo gerente. Quando o agente recebe um comando SNMP, este faz a atualização ou consulta da MIB e, assim, atua sobre a simulação, alterando os arquivos de entrada do SITRA-B+.

Esta aplicação utiliza a linguagem de programação C++, em um ambiente Windows, para o desenvolvimento do gerente e do agente SNMP. Porém, o processo agente pode ser construído com as mesmas características em outro sistema operacional, como, por exemplo, LINUX. Somente o gerente SNMP, devido às restrições impostas pelo SITRA-B+, deve ser compatível com o sistema operacional Windows.

A interface do gerente SNMP consiste de uma janela gráfica, contendo uma linha de comando e quatro botões, como mostra a Figura 5.7.

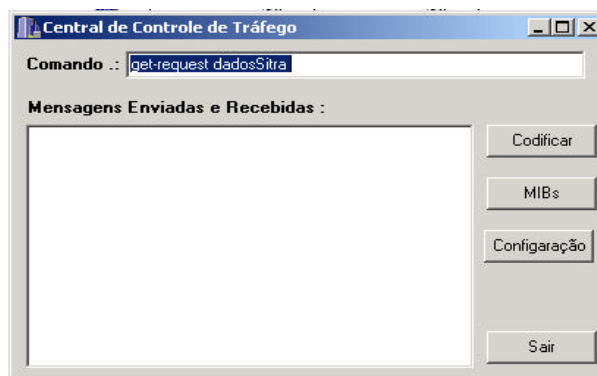


Figura 5.7 – Interface gráfica dos processos Gerente e Agente.

Ao ser ativado, o gerente requisita, ao agente SNMP o estado atual da microssimulação, executa seu algoritmo de controle e, se necessário, envia o novo estado dos semáforos para o

agente SNMP. A linha de comando permite que o operador digite um comando SNMP e o envie ao agente pressionando o botão codificar. O (botão MIBs) mostra, na janela, o estado local dos objetos de dados e o (botão Configuração), os dados relativos à conexão, como IP local, porta e outros.

A interface do agente SNMP serve somente para visualizar a comunicação entre o gerente e o SITRA-B+. Esta consiste de uma janela com dois botões: um para mostrar o estado local dos objetos de dados, (botão MIBs); e outro, para os dados relativos à conexão, (botão Configuração), exatamente como no processo gerente SNMP. A Figura 5.8 mostra a janela do agente SNMP da aplicação.

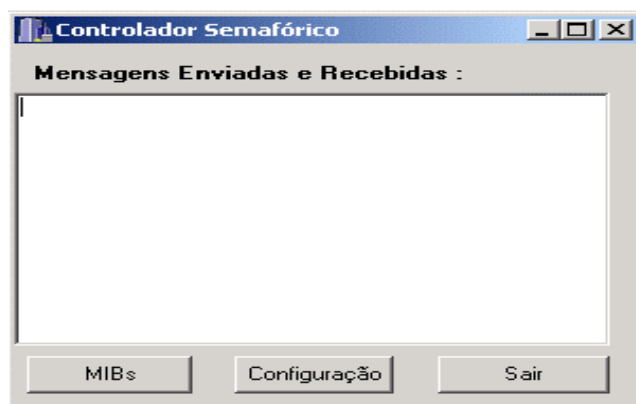


Figura 5.8 – Interface gráfica dos processos Gerente e Agente SNMP.

Apesar aplicações gerente e agente não requerem interface com o usuário, já que o controle dos dispositivos se dá pela avaliação do estado das vias por um algoritmo de controle de tráfego. Mesmo assim, projetou-se uma interface que permite ao operador atuar diretamente sobre o estado dos semáforos ou consultar os objetos de dados em um instante qualquer. Esta possibilidade de intervenção no sistema está presente na maioria dos SIT, já que os algoritmos são incapazes de solucionar todos os impasses gerados pela demanda do tráfego, requerendo a atuação humana no sistema.

5.6. O Ambiente de Simulação

O software SITRA-B+ (SODIT, 2002) oferece um ambiente de microsimulação que permite consultar ou atuar sobre as variáveis da simulação em tempo real. Para a simulação de tráfego, pode-se definir em detalhes o cenário desejado, ou seja, as características das vias, largura, número de pistas, tipos de veículos (automóveis, caminhões, ônibus, motos), movimentos possíveis nas interseções, composições dos fluxos em termos de taxas de conversão, etc.

O SITRA-B+ pode operar com controle fixo ou a partir de uma estratégia externa. O controle externo dá-se com a alteração um conjunto de variáveis que representam o estado dos dispositivos semafóricos e estão armazenadas em arquivos texto. Os dados sobre o estado das vias são, também, coletados de arquivos do tipo texto, atualizado pelo simulador a cada passo da microsimulação.

A aplicação atua sobre um ambiente virtual, formado por um conjunto de oito detectores veiculares e dois indicadores semafóricos. O Anexo 6 traz uma breve descrição das funcionalidades deste simulador de tráfego.

Apesar da aplicação operar sobre este ambiente virtual existem exemplos reais de aplicativos com características tecnológicas semelhantes. São sistemas de controle de tráfego e mensagens dinâmicas, utilizados em Toronto, Canadá, que usam SNMP sobre UDP/IP e *Ethernet*, sobre uma mistura de tecnologias de comunicação (FHWA, 1999d).

5.7. Considerações Finais

A implementação deste aplicativo de comunicação tornou possível a observação das demandas e limites tecnológicos do NTCIP, além de seu alcance e restrições operacionais. Um ponto importante é que, a aplicação aqui implementada serve também como orientação para o desenvolvimento de novos produtos, facilitando o processo de levantamento tecnológico de padrões, requisitos e restrições do NTCIP em outros domínios de aplicações.

Apesar do grande número de documentos, textos e sítios web sobre o assunto, a abordagem dada ao tema é, na sua maioria, superficial, quanto às questões do desenvolvimento de um produto real. Não foi encontrado nenhuma referencia bibliográfica ou aplicativo NTCIP já implementado. Esta falta de referência dá a entender que, apesar de se tratar uma proposta baseada em tecnologias abertas, sua aplicação não seria trivial.

São questões como estas que, talvez, tenham intimidado o uso de protocolos padrão por parte dos desenvolvedores de produtos SIT nacionais. Entretanto, uma vez reconhecidas às tecnologias e resolvidas as questões de adaptabilidade destas ao perfil tecnológico nacional, sua implementação passa a ser tão complexa quanto à de qualquer outro produto de software comercial.

A arquitetura centralizada do SNMP, aqui utilizada atende perfeitamente ao método de operação dos sistemas SIT. No qual uma central controla dispositivos junto à pista. A comunicação funcionou sem problemas sobre a rede local. Observa-se, pela Figura 5.6, que as mensagens são da ordem de poucas centenas de caracteres (ou bytes). A título de ilustração da aplicabilidade prática do protocolo, seja o caso de uma comunicação serial a 1200 bytes/s, típica para equipamentos de campo. Uma mensagem de 300 bytes implica no tempo de envio de $300/1200 = 0,25s$. Uma central de tempo real troca dados com os dispositivos de campo a cada 5s. Portanto, o tempo de comunicação é adequado. Entretanto, os proponentes do NTCIP definiram um protocolo derivado do SNMP, chamado de STMP (para *Simple Transportation Management Protocol*), que reduz por um fator de 10 a 15 o tamanho das mensagens trocadas. Este é totalmente compatível com sistemas SNMP, ou seja, tem-se a interoperabilidade dos equipamentos STMP e SNMP.

Porém, a idéia de utilizar-se unicamente o SNMP para controlar um SIT não é a melhor solução, o protocolo adapta-se melhor em sistemas de tráfego com pequeno número de equipamentos gerenciados. Já em SIT maiores, com muitos dispositivos e grande tráfego de dados sobre o canal de comunicação, o protocolo tende a consumir muitos recursos da rede.

6. CONCLUSÕES E PERSPECTIVAS FUTURAS

O reconhecimento das tecnologias e padrões envolvidos na implementação de um dos protocolos do nível de aplicação e da formalização dos objetos de dados manipulados pelo NTCIP, descritos no Anexo 1, mostra que a proposta busca a simplicidade em suas estruturas de comunicação. Isto não significa que os métodos empreendidos neste sentido sejam triviais ou com alcance limitado, ao contrário, a proposta é muito abrangente e sua utilização demanda um minucioso estudo de todos os padrões. Assim como a NIA descreve uma arquitetura para a construção de SIT, o NTCIP também é uma referência para que se possa atingir a interoperabilidade e intercambialidade de dispositivos e sistemas nestes SIT.

Analisando objetivamente a infraestrutura de comunicação descrita pela proposta do NTCIP e isentando-se das descrições comerciais do produto, pode-se concluir que o uso desta tecnologia viabilizaria, em curto prazo, a integração de qualquer SIT. Em suma, o uso de dispositivos com tecnologia NTCIP, que inicialmente poderiam operar conjuntamente com equipamentos já existentes (ou “sistemas legados”), em um futuro próximo permitiriam a criação de uma malha nacional de SIT.

A tecnologia NTCIP implementada em parte através do protocolo SNMP mostrou-se ser adequada para o armazenamento, o gerenciamento e a comunicação de dados, fornecendo transparência de produto e tornando indiferente o reconhecimento das tecnologias de cada fornecedor. Este protocolo mostrou que é de fácil entendimento e implementação, apresentando boa flexibilidade de utilização, além de permitir ao desenvolvedor utilizar seus critérios tecnológicos, desde que mantidos os padrões de comunicação.

Com referência à realidade nacional, conclui-se pela viabilidade no uso da pilha de protocolos utilizada nesta aplicação NTCIP, no que tange aos custos dos canais de comunicação com capacidade para as mensagens trocadas, pois estes podem ser construídos com pares de fios telefônicos convencionais, ou mesmo usarem a infra-estrutura das companhias telefônicas e as tecnologias empregadas, que são baseadas em protocolos públicos que fundamentam a operação de redes Internet.

É importante destacar que, em sua grande maioria, os desenvolvedores possuem familiaridade com os conceitos usados no NTCIP. Em particular, a aplicação ora descrita usa padrões largamente empregados nas gerências de redes Internet, como SNMP, TCP/IP, Ethernet e outros. Assim, um padrão nacional baseado em NTCIP não traria dificuldades para os fabricantes, no sentido da adaptação dos controladores semafóricos existentes. Mesmo que se limite a discussão ao controle semafórico, o benefício para a coletividade seria imediato, por três motivos:

- Do ponto de vista econômico, a dependência comercial de um fornecedor de equipamentos SIT seria reduzida, graças a intercambialidade e interoperabilidade

gerada pela comunicação dos dispositivos, representando o fim das reservas de mercado hoje existentes;

- Do ponto de vista da independência tecnológica, surge à possibilidade do desenvolvimento de um padrão nacional, baseado em uma arquitetura e modelo de comunicação testados e com alto grau de amadurecimento. Isto permitiria que, em um futuro próximo, saíssemos da condição de consumidores de produtos e dispositivos SIT, para a de fornecedores a um mercado mundial e em constante expansão.
- Do ponto de vista da engenharia de tráfego, abre-se à possibilidade da implantação gradativa de soluções avançadas de controle de tráfego urbano. Assim, por exemplo, uma municipalidade pode investir em controladores que operem em rede para, só mais tarde, vir a adquirir uma central de controle de tráfego. Esta pode ser apenas uma central de comando dos controladores semafóricos, tanto para monitoração do estado como para programação remota destes, passando para uma central de gerência de tráfego e até de controle em tempo real.

Em relação às perspectivas futuras deste trabalho, pode-se dizer que se neste momento, o arcabouço tecnológico envolvido na implementação de um dos protocolos do NTCIP foi esclarecido, existem outros que ainda estão velados sob uma documentação desconexa ou em desenvolvimento. Diversas vezes, neste trabalho, foi citado que tanto a NIA quanto o NTCIP são referências e encontram-se em desenvolvimento e adaptação. Esta adaptação se dá inicialmente ao mercado Norte Americano, o que não nos impede de construir uma versão nacional, adequada às nossas peculiaridades de tráfego e tecnológicas.

Assim, se o primeiro passo na elucidação do que é o NTCIP foi dado, muito ainda deve ser realizado para que possamos definitivamente atingir a independência tecnológica em uma área tão importante, como a do controle de tráfego.

7. REFERÊNCIAS BIBLIOGRÁFICAS

AAHSTO-NTCIP - **Battelle Memorial Institute AAHSTO-NTCIP. AAHSTO-NTCIP - Octet Encoding Rules(1102), AAHSTO-NTCIP - Simple Transportation Management Protocol (1103).** Março, 2003 http://www.public.bdo.battelle.org/webprojs/its/Tr_0/App3/AAHSTO-NTCIP.html.

Araújo, A.P. e Kraus Jr., W. (2002). **Sistema de Informação em Transportes Baseado na National ITS Architecture.** XVII Congresso da Associação Nacional de Pesquisa e Ensino em Transportes. Ed. ANPET, Natal, RN.

Bass, L., Clements, P., Kazman R. (1998) - **Software Architecture in Practice**, Addison Wesley.

Bonetti, W. J. (2001) – **Utilização e Parametrização de Semáforos Atuados pelo Tráfego** – Escola Politécnica da Universidade de São Paulo, Departamento de Engenharia de Transportes, São Paulo, Brasil.

Cambruzzi, E., e Kraus Jr., W. (2003). **Sistema de Informação em Transportes Baseado na National ITS Architecture.** XVII Congresso da Associação Nacional de Pesquisa e Ensino em Transportes. Ed. ANPET, Rio de Janeiro, RJ.

Commer D. C. , Stevens, D. L., (1992)- **Internetworking with TCP/IP - Design Implementation and Internals, Volume II**, Prentice Hall Segunda Edição, USA.

Commer, D. C., (1991) - **Internetworking with TCP/IP - Principles, Protocol and Architecture, Volume I**, Prentice Hall Segunda Edição, USA.

FHWA - Federal Highway Administration (1997a) – **ITS Communication Document** - US Department of Transportation – Washington D.C., EUA.

FHWA - Federal Highway Administration (1997b) – **Global Objects Definitions** - US Department of Transportation – Washington D.C., EUA.

FHWA - Federal Highway Administration (1998) – **Real World Benefits** – US Department of Transportation – Washington D.C., EUA.

FHWA - Federal Highway Administration (1999a) – **ITS Logical Architecture – Volume I** – US Department of Transportation – Washington D.C., EUA.

FHWA - Federal Highway Administration (1999b) – **ITS Physical Architecture – Volume I** – US Department of Transportation – Washington D.C., EUA.

- FHWA - Federal Highway Administration (1999c) – **Executive Summaries** – US Department of Transportation – Washington D.C., EUA.
- FHWA - Federal Highway Administration (1999d) – **The NTCIP Guide/National Transportation Communication for ITS – Protocol (NTCIP)** – *Joint Committee on the NTCIP* – Washington D.C., EUA.
- FHWA - Federal Highway Administration (2002a) - **National Transportation Communications for ITS Protocol (NTCIP) – Data Element Definitions for Transportation Sensor Systems** – US Department of Transportation – Washington D.C., EUA
- FHWA - Federal Highway Administration (2002b) - **National Transportation Communications for ITS Protocol (NTCIP) – Transportation Management Protocols** – US Department of Transportation – Washington D.C., EUA
- FTE - **Frontline Test Equipment: Products - Intelligent Transportation Systems** – Novembro, 2003, <http://www.fte.com/prdntcip.asp> .
- Hernán A., Stuart H. (1998) - **Understanding the Architect’s Job**, Software Architecture Workshop of OOPSLA’98.
- Hills P., Rhodes, T. Allsop, R. (1987) – **Road and Traffic in Urban Areas, Institution of Highways and Transportation**, Department of Transport.
- Homburger, W. S., Kell, J. S., Perkins D. D. (1992) – **Fundamentals of Traffic Engineering** – 13th ed. Institute of Transportation Studies, University of California - Berkley, USA.
- IAB - Internet Architecture Board - (1993b) STD 16 [RFC 1155] – **Estrutura e Gerenciamento de Informação baseado em TCP/IP** .
- IAB - Internet Architecture Board - (1993c) STD 15 [RFC 1157]– **SNMP (Simple Network Management Protocol)**.
- IAB - Internet Architecture Board - (1993d) STD 17 [RFC 1223] – **Management Information Base para Redes de Gerenciamento**.
- IAB - Internet Architecture Board - (1997) STD 17 [RFC 1212] – **Concise MIB (Management Information Base)**.
- IAB - Internet Architecture Board (1993a) - STD 15 [RFC 1157] – **SNMP (Simple Network Management Protocol)**.

Information Processing Systems, (1987) - **Open Systems Interconnection, Specification of Basic Encoding Rules for Abstract Notation One ASN.1**, International Organization for Standardization International Standard 8825.

ISO/IEC [8825-1] (1998) – **ASN.1 (ASN.1 Encondig Rules, Especificação das Basic Encondig Rules - BER)**.

ITS-Things to Consider" When Adopting NTCIP – Abril, 2003, <http://www.itsa.org/committe.nsf>.

ITU-T Recommendation X.690 (1997) - **OSI networking and system aspects – Abstract Syntax Notation One (ASN.1) - Information technology –ASN.1 encoding rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER)** –Telecommunication Standardization Sector of ITU.

Jazayeri, M., Ran, A., Linden, V. D. F. (2000) - **Software Architecture for Products Families**, Addison Wesley.

National Electrical Manufactures Association – NEMA – Virtual Documentation – Outubro, 2003. <http://www.nema.org>.

National Transportation Communications for ITS Protocol – NTCIP – **Documents** – Maio, 2003, [http:// www.ntcip.org/library/documents](http://www.ntcip.org/library/documents).

NEMA Standards Publication TS 3.2 (1996) - **National Transportation Communications for ITS Protocol (NTCIP) - Simple Transportation Management Framework** –Washington D.C., EUA.

NTCIP C2F Standard Activities Group –Março, 2003, http://www.itsarch.iteris.com/ isarch/ html/ standard/ntcipc2f_b.htm.

Projeto SincMobil – **Notícias** – <http://www.lcmi.ufsc.br/sincmobil/noticias.html>.

Rose, M. (1998) - **Management Information Base for Network Management of Information for TCP/IP – based Internets. RFC 1155**, maio 1990.

SODIT, Manuel Utilisateur (Março 2002) – **Logiciel SITRA. SODIT**, Toulouse, França.

Stallings, W. (1999). **SNMP, SNMPv2, SNMPv3 and RMON 1 and 2**. Addison Wesley, 3° ed..

Szasz, P.A. (1997) - **Estudo de Viabilidade de Implantação de Centralização Semafórica na Cidade de Campinas** – Empresa Municipal de Desenvolvimento de Campinas (EMDEC), Campinas, Brasil.

Tanenbaum, W. S. (1994) – **Redes de Computadores** – Editora Campus – Rio de Janeiro, RJ.

Telematics Application Programme (1999) – **KAREN Physical Architecture, European Communities** – European Community

Anexo 1

PADRÕES PARA COMUNICAÇÃO UTILIZADOS NA *NATIONAL ITS ARCHITECTURE*

1. Introdução

O tratamento da representação dos dados é uma das maiores dificuldades operacionais entre sistemas de rede abertos. A solução para problemas de representação, codificação, transmissão e decodificação de estruturas de dados é ter um modo de descrever estas estruturas que seja suficientemente flexível para ser utilizado em várias aplicações (Tanenbaum, 1994).

Buscando uma solução para estas questões, a NIA reuniu seus esforços de padronização de protocolos sob o nome NTCIP - *National Transportation Communications for ITS Protocol*. O NTCIP não é um protocolo, e sim um conjunto de protocolos para as várias camadas do modelo OSI, muitos dos quais já largamente usados em outros domínios. Ele permite que diferentes fabricantes desenvolvam produtos SIT com as características de interoperabilidade definidas na arquitetura.

A proposta do NTCIP define um conjunto de estruturas padrão para a descrição, armazenamento, codificação e transmissão de pacotes de dados entre dispositivos SIT. Além disso, para que computadores troquem mensagens entre si, é necessário que exista um meio físico (canal de comunicação) e, principalmente, um conjunto de regras (que constituem protocolo de comunicação), que tratem os erros, falhas e impasses. O conjunto de regras para a descrição e codificação adotado pelo NTCIP é definido na *Abstract Syntax Notation One - ASN.1*, esta notação é a base para a maioria das trocas de dados do protocolo e, portanto, assunto de nossas próximas seções.

2. Aspectos Gerais da *Abstract Syntax Notation One - ASN.1*

A idéia básica da *Abstract Syntax Notation One* é definir todos os tipos de estruturas de dados exigidos por qualquer aplicação que utilize a notação e agrupá-los em um módulo biblioteca. Assim, usando a ASN.1 como guia, protocolos definidos sobre a camada de aplicação (modelo OSI/ISO), passam a estrutura dos dados que serão transmitidos para a camada inferior, que reconhece os tipos e tamanhos dos dados, os codifica e transmite (Tanenbaum, 1994). No outro extremo da rede, a camada de apresentação receptora observa a identidade dos dados ASN.1 e decodifica os dados, como ilustra a Figura 1.

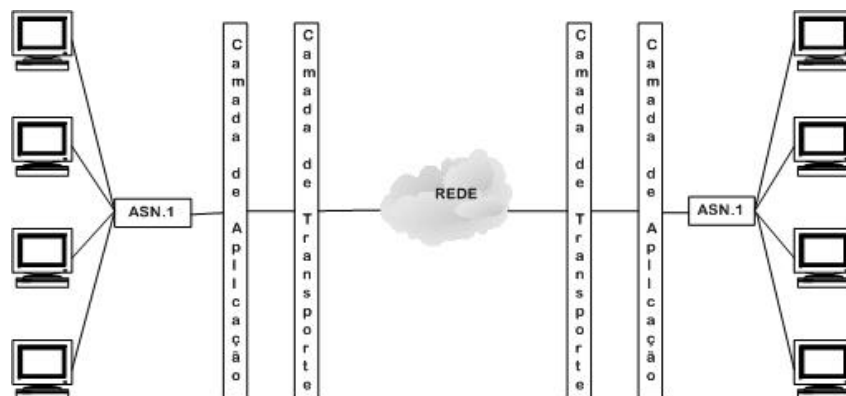


Figura 1 - Esquema simplificado entre transmissor e receptor usando ASN.1.

A notação ASN.1 tem sido utilizada para a definição de *Package Data Units* – PDUs, em protocolos implementados sobre a camada de aplicação tanto da ISO quanto da INTERNET, tais como o SNMP, DATEX-ASN e outros.

Nesta seção, estão descritas as principais estruturas da notação, sua sintaxe, construtores e tipos. Não existe a pretensão de fornecer uma completa revisão desta notação, mas, sim, de elucidá-la, de forma a permitir sua compreensão e quais são suas implicações na implementação do NTCIP.

2.1. Tipos Primitivos

Os tipos primitivos são definidos como palavras reservadas da notação e devem ser escritos integralmente e em letra maiúscula.

- **INTEGER:** São números ordinários utilizados para contagem e não possuem tamanhos máximos. A notação permite definir nomes para os valores, como nos exemplos abaixo:

decimal ::= **INTEGER** { um(1), dois(2), três(3), quatro(4)}, assim o identificador “um” está ligado ao valor “1”.

intervalo ::= **INTEGER** { min(10), max(100)}, onde os valores 10 e 100 são associados aos identificadores min e max respectivamente.

Além destes, outros exemplos de valores são: 100, 234, -1999, 3763432653645634726372643532.

- **OCTET STRING:** É uma cadeia de bytes e pode representar qualquer tipo de dados, orientado a byte. Além de não possuir tamanho máximo determinado.
- **BIT STRING:** Trata-se de uma cadeia de bits sem tamanho definido ou com a necessidade de ser formada por múltiplos de oito. Sua utilização se dá principalmente na representação de bits nomeados, como no exemplo abaixo:

```
portaAtiva::= BIT STRING { LPT1(1),
                           COM1(2),
                           COM2(3),
                           USB(4) }
```

Neste caso, apenas um dos itens será validado, os demais serão inicializados com zero. Outros exemplos de valores são:

"01001010100010100010" (cadeia de bits)

"A0FF" (cadeia de hexadecimais)

- **BOOLEAN:** Utilizado para representar somente dois estados (TRUE ou FALSE).
- **NULL:** Representa a inexistência de valor.
- **OBJECT-IDENTIFIER:** Cada tipo de objeto, denominado *Object-Type*, possui um nome, uma sintaxe e uma codificação. O nome é representado univocamente como um IDENTIFICADOR DE OBJETO (*Object Identifier*). Os identificadores de objetos são um nome administrativo determinado e são mais do que uma seqüência de octetos hexadecimais dispostos lado a lado; são uma estrutura de endereço que atravessa uma árvore global onde cada octeto representa um nó dessa árvore. Nesta estrutura, o nó raiz não é rotulado, mas possui ao menos três sub-árvores abaixo dele, conforme exemplo a seguir:

nome **OBJECT IDENTIFIER**::= { OSI(1).ORG(3).DOD(6). ... }

Cada inteiro representa um endereço e é separado de seu sucessor por um ponto, formando assim um conjunto que, ao final, identificará o objeto a ser codificado, como por exemplo:

nema **OBJECT IDENTIFIER**::= { OSI(1).ORG(3).DOD(6).
directoty(). mngnt().private().enterprises().user... }

Assim o endereço de um dispositivo dentro do padrão NEMA poderia ser reescrito utilizando-se ASN.1 da seguinte forma:

nema **OBJECT IDENTIFIER**::= { OSI(1).ORG(3).DOD(6).
INTERNET(1). PRIVATE(4).ENTERPRISE(1). NEMA(1206)... }

ou apenas

nema **OBJECT IDENTIFIER**::={1.3.6.4.1.1206. ... }

- **OBJECT-TYPE:** Representa o nome textual de um objeto (*Object Descriptor*) o qual acompanha seu correspondente identificador de objeto (*Object Identifier*). Este objeto é composto por quatro campos:

SYNTAX: usada para definir a que tipo de estrutura de dados pertence o objeto, feito sempre, utilizando a linguagem formal ASN.1.

ACCESS: Define como a base de dados será acessada: *Read-only, Read-write, Not-accessible*

STATUS: Define se este objeto é requerido:

Mandatory: o objeto é suportado e aparece na descrição do padrão;

Optional: o objeto pode não ser suportado;

Decrepate: o uso deste objeto é desencorajado;

Obsolete: o objeto pode vir a ser substituído ou excluído em novas versões.

DESCRIPTION: É a descrição textual da semântica do objeto, esta deve ser clara e o objeto deve efetuar exatamente o que está proposto na descrição.

O exemplo abaixo descreve o objeto (OBJECT-TYPE) placaDoVeículo, pertencente a um identificador de objetos (OBJECT-IDENTIFIER) fictício, usando a notação ASN.1:

```
placaDoVeiculo OBJECT-TYPE
SYNTAX INTEGER
ACCESS read-only
STATUS mandatory
DESCRIPTION "Contém somente o número da placa do
                veículo."
 ::= { fictício }
```

2.2. Tipos Construídos

- **SEQUENCE:** O tipo construído SEQUENCE é utilizado para combinar uma coleção de tipos ASN.1, gerar listas ou tabelas de elementos onde a ordem é significativa. Como no exemplo abaixo:

```
Departamento ::= SEQUENCE { nome VisibleString,
                             laboratorio VisibleString,
                             sala INTEGER }
```

- **SEQUENCE OF:** O tipo SEQUENCE OF é utilizado para a construção de tabelas bidimensionais, estas tabelas são formadas por tipos primitivos ou SEQUENCE.

```
SEQUENCE OF EventClassEntry
    EventClassEntry ::= SEQUENCE {
        eventClassNumber INTEGER,
        eventClassLimit INTEGER,
        eventClassClearTime Counter,
        eventClassDescription OCTET STRING,
        eventClassNumRowsInLog INTEGER }
```

- **SET:** Este tipo é muito semelhante ao SEQUENCE, a diferença é que a ordem dos componentes não é significativa.

```
cadastro ::= SET { nome [0] VisibleString;
                 cpf [1] integer }
```

- **SET OF:** Possui a mesma funcionalidade do SEQUENCE OF com a particularidade de que a ordem dos elementos não é significativa.

```
cores ::= SET OF VisibleString
```

- **CHOICE :** O tipo CHOICE é utilizado quando uma estrutura de dados pode conter vários tipos diferentes assemelhando-se ao SEQUENCE, como podemos observar no exemplo a seguir:

```
nome ::= CHOICE { nomeCliente OCTET STRING,
                 codigoCliente INTEGER,
                 classeCliente COUNTER }
```

- **ANY:** Este tipo é utilizado para modelar uma variável não especificada ou especificada em outro ponto.
- **EXTERNAL:** É usado para referenciar um tipo não especificado ou descrito em outro local.

2.3. Atributos Especiais de Valores

A ASN.1 define duas estruturas especiais que permitem ao receptor atribuir um tipo ou um valor, a um dado que não esteja codificado no pacote recebido. São elas:

- **Optional:** Utilizada quando o tipo não existir, isto significa que o tipo pode ou não ser transmitido no pacote.
- **Default:** Este atributo é seguido sempre de um valor associado, que é assumido quando o receptor recebe um pacote sem um valor definido.

2.4. Rótulos

Os rótulos na notação ASN.1 são utilizados para a identificação dos tipos ou campos e são compostos por duas partes, classe e número. As classes são quatro: *APPLICATION*,

UNIVERSAL, *PRIVATE* e *CONTEXT ESPECIFIC*. Já o número é representado por um inteiro positivo.

A função dos rótulos é a de relacionar a sintaxe de transferência com as regras de codificação. Assim, toda vez que um item é transmitido, o tipo, tamanho e valor deste são enviados na forma de octetos ao receptor. As classes correspondem a:

- **UNIVERSAL:** Utilizada para os tipos primitivos e construtores da notação ASN.1;
- **APPLICATION:** Esta para tipos definidos dos protocolos de aplicação da ISO;
- **PRIVATE:** Para tipos definidos pelo usuário;
- **CONTEXT SPECIFIC:** Utilizada para distinguir tipos diferentes dentro de um mesmo pacote.

Por si só a ASN.1 não fornece ferramentas para a codificação dos dados a serem enviados em canal de comunicação. Para isso, a NIA utiliza-se de vários conjuntos de regras de codificação e decodificação definidos pela ISO. Um desses conjuntos de regras é denominado, *Basic Encoding Rules* – BER e é usado em conjunto com o protocolo SNMP, ambos descritos nas seções seguintes.

3. O Padrão de Codificação *Basic Encoding Rules* – BER

As regras de codificação BER fornecem um algoritmo para que um valor de qualquer tipo ou estrutura definida usando-se ASN.1, possa ser codificado para transmissão. Transmissor e receptor utilizam-se do mesmo algoritmo para codificação e decodificação dos bits. É evidente que o receptor deve conhecer as definições dos tipos e estruturas para identificar através do rótulo qual o tipo recebido. A codificação de um tipo de dado ASN.1 usando BER, permite que o receptor reconheça o início e o fim das estruturas e os tipos e valores contidos nas seqüências de octetos. Estas seqüências são denominadas TLV (*Tag-Length-Value*) e uma mensagem ou *frame* transmitida pode conter uma ou mais seqüências do tipo TLV. A Figura 2, ilustra a estrutura TLV definida pela ASN.1 e usada pelo NTCIP em seu protocolo *Center-to-Field*.

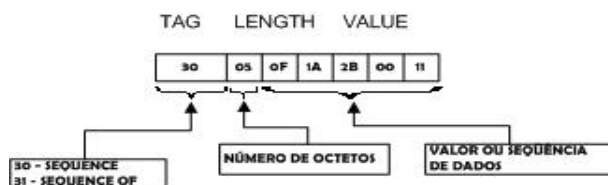
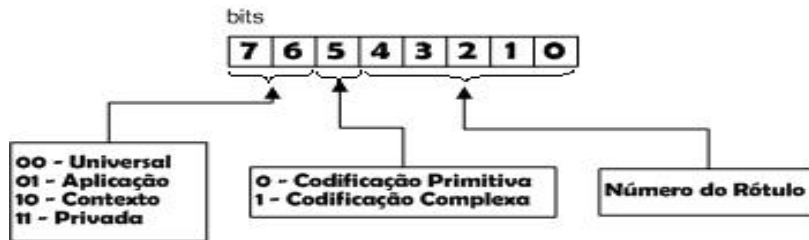


Figura 2 - Representação de um frame de dados usando ASN.1.

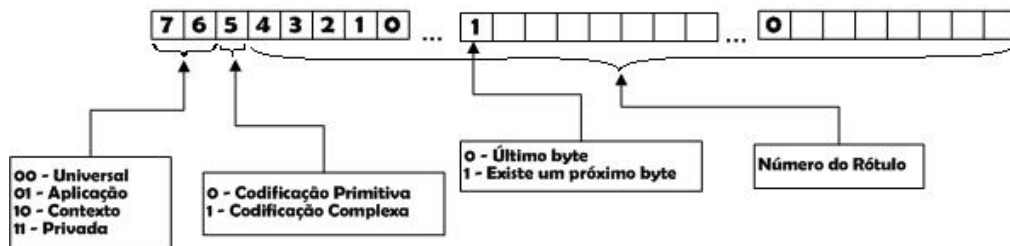
3.1. Regras Gerais de Codificação

Para a codificação de valores dos tipos descritos pela ASN.1, com exceção dos externos, os dados devem aparecer na seguinte ordem:

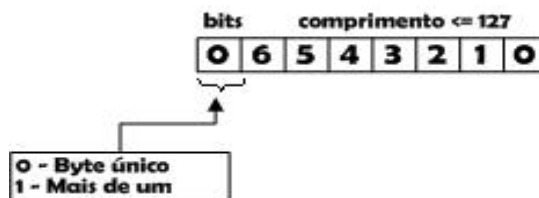
- **Octetos de Identificação**



No caso do número do rótulo necessitar mais do que cinco bits, utiliza-se então o seguinte formato:



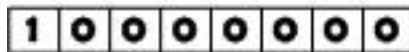
- **Octetos de Comprimento:** Para a codificação de octetos de comprimento inferior a 127 um único byte é suficiente; para valores maiores, podem ser agregados n bytes.



O exemplo a seguir ilustra como são codificados comprimentos maiores que 127 bytes:



O último octeto da série é um sinalizador de fim de conteúdo e deve ser descrito da seguinte forma:



- **Octeto de Conteúdo:** Os octetos de conteúdo são codificados segundo o tipo e características associadas a cada um, ou seja, inteiros podem ser codificados de diferentes formas, assim como as *strings* podem ser representadas diretamente ou codificadas em octetos hexadecimal. As normas para a codificação de cada um dos tipos encontram-se detalhadas na recomendação X.690 12/97 da CCITT.
- **Octeto de Fim-de-Conteúdo:** Este octeto é utilizado somente quando o comprimento do conteúdo que está sendo transmitido não é conhecido. Nesse caso, o octeto de fim-de-conteúdo serve como delimitador da mensagem.

Reconhecidos os métodos de descrição e codificação para transmissão dos dados no NTCIP, resta definir os critérios de gerência e armazenamento destes. Como solução, a NIA define que as informações disponibilizadas pelos dispositivos em sua arquitetura SIT devem obedecer à estrutura de armazenamento da *Management Information Base* – MIB e estarem em concordância com as normas utilizadas pela *Internet Activities Board* – IAB.

4. Princípios da *Management Information Base* – MIB

A MIB, é a descrição conceitual da informação a ser gerenciada, incluindo objetos e os atributos associados a estes. Os principais padrões são a MIB da OSI e a MIB da Internet, sendo que a segunda é a utilizada dentro da proposta de arquitetura de referência da NIA.

Os objetos gerenciados MIB foram publicados em 1989 sob o nome de MIB-I. Em 1990, sofreram uma revisão e passaram-se a se chamar MIB II e em 1993 foram acrescentadas extensões às definições dos objetos.

Operacionalmente um objeto na MIB representa um recurso do dispositivo sendo que cada identificador de objeto (*object-identifier*) é a instância de uma variável de ambiente. Portanto, o conjunto desses objetos define o estado de um dispositivo e a manipulação destes provoca uma alteração neste dispositivo. Além disso, o conceito de identificador de objetos traz agregado a noção de hierarquia e identifica um objeto sem levar em consideração seu significado. Desta forma,

um objeto na MIB, pode representar não somente estados (dados), mas também padrões ou órgãos regulamentadores.

Um identificador de objetos é uma seqüência de inteiros que pertence a uma árvore global, esta árvore consiste de uma raiz conectada a um número de nós rotulados lado a lado. Cada nó pode desse modo ter seus próprios filhos, os quais são rotulados, formando sub-árvores. Este processo pode continuar de um nível arbitrário ao mais profundo.

O identificador pode também ser descrito como um controle administrativo dos significados dos nós em árvore global abstrata com uma raiz anônima. Os níveis desta árvore são compostos pelos nós de dados individuais identificados pelos *Objects Identifiers* - OID que nomeiam univocamente os objetos da MIB na árvore.

Um OID é representado por uma cadeia de números separados por pontos, organizados de forma hierárquica com um dos dígitos associado a uma organização e após este uma sub-árvore que obedece a mesma regra. A Figura 3, mostra a estrutura da árvore MIB onde a *National Electrical Manufactures Association* - NEMA aloca um nó sob a categoria *enterprises*.

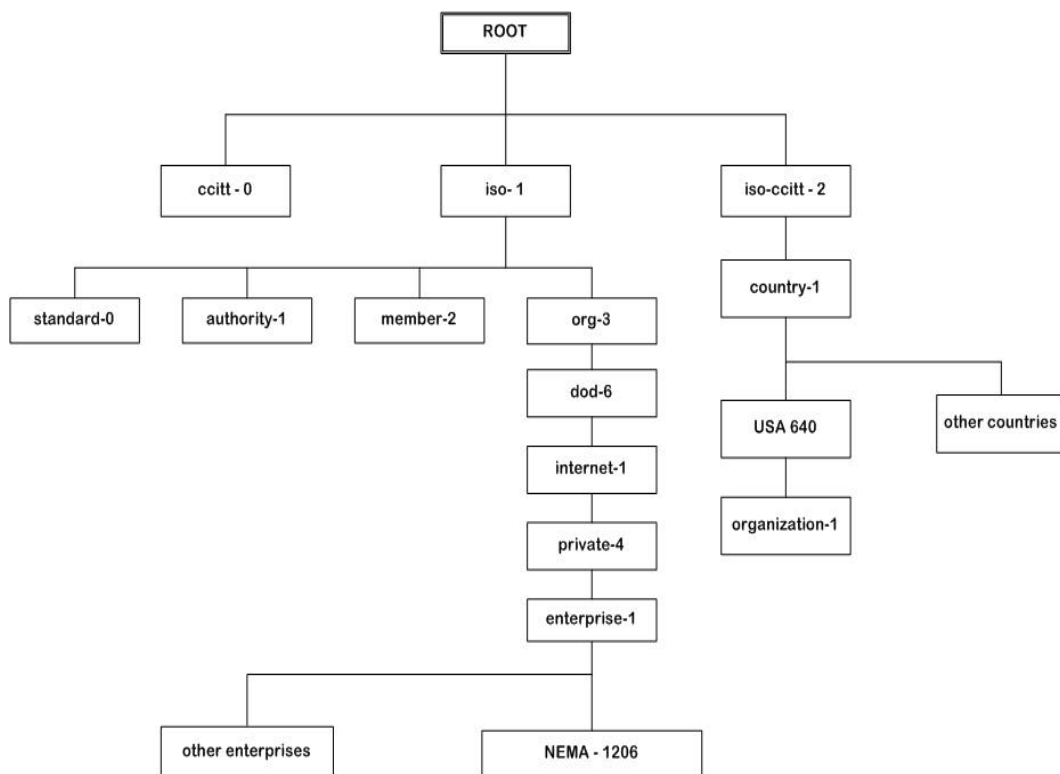


Figura 3 - Estrutura da árvore de objetos MIB (FHWA, 1999d).

A MIB da Internet define os objetos que podem ser gerenciados pelo protocolo TCP/IP e é formada por um nó raiz não rotulado e tem pelo menos três nós diretamente abaixo dele. O primeiro, é o nó administrado pela ISO(1) - *International Organization for Standardization*. O segundo é o nó do CCITT(0), *International Telegraph and Telephone Consultative Committee* (atual ITU-T) e o terceiro é administrado em conjunto pela ISO e CCITT, denominado *joint-iso-ccitt(2)*.

Abaixo do nó ISO(1) designou-se uma sub-árvore para outras instituições denominado *organization(3)* ou *ORG(3)*. Este nó foi utilizado pelo *National Institutes of Standards and Technology* – NIST que o cedeu ao departamento de defesa dos Estados Unidos o *Department of Defense* - DOD, denominado DOD(6). O DOD não definiu sua estrutura de objetos na árvore, ficando então este nó como da comunidade Internet, administrado pela IAB.

Sob o nó Internet, são especificados quatro possíveis filhos:

- `directory OBJECT IDENTIFIER ::= { Internet 1 }`
- `mgmt OBJECT IDENTIFIER ::= { Internet 2 }`
- `experimental OBJECT IDENTIFIER ::= { Internet 3 }`
- `private OBJECT IDENTIFIER ::= { Internet 4 }`

directory(1): Esta sub-árvore é reservada para uso da ISO na Internet;

mgnt(2): Utilizada para identificar objetos aprovados pela IAB. Significa que os objetos foram aprovados e certificados;

experimental(3): Utilizado para identificar objetos que estão em experiência na Internet;

private(4): A sub-árvore *private(4)* é utilizada para identificar objetos definidos unilateralmente. Esta sub-árvore tem no mínimo uma folha.

`enterprises OBJECT IDENTIFIER ::= { private 1 }`

A sub-árvore *enterprises(1)* permite que desenvolvedores de sistemas de rede registrem seus modelos de produtos. Ao receber um nó, o desenvolvedor registra seus objetos MIB abaixo deste, evitando assim ambigüidades na identificação dos objetos na árvore. A NEMA aloca o nó 1206 e a partir deste, define sua estrutura de objetos como se pode observar na Figura 4.

Os objetos contidos em uma MIB são acessados e monitorados por um agente de gerenciamento e a comunicação entre estes pode ser feita usando o protocolo *Simple Network Management Protocol* - SNMP. O SNMP é um protocolo de uso genérico amplamente difundido na Internet e utilizado pela NIA como um dos padrões de comunicação do NTCIP, para a troca de dados entre dispositivos SIT.

5. Componentes do *Simple Network Management Protocol* - SNMP

O *Simple Network Management Protocol*, é um protocolo da camada de aplicação, usado para facilitar a comunicação entre um gerente e dispositivos agentes em uma rede. Este protocolo utiliza objetos MIB e opera sobre *User Datagram Protocol / Internet Protocol* - UDP/IP, onde não existe controle de conexão entre a estação de gerenciamento e o agente gerenciado. Pode-se definir que o modelo de gerência SNMP, engloba um protocolo de transmissão de informações entre um ou mais sistemas gerentes e seus agentes, em uma estrutura para especificação e armazenamento de informações de gerenciamento (Stallings, 1999).

A Figura 5 mostra a pilha de protocolos necessária para a comunicação entre uma estação gerente e um dispositivo agente, utilizando o protocolo SNMP .

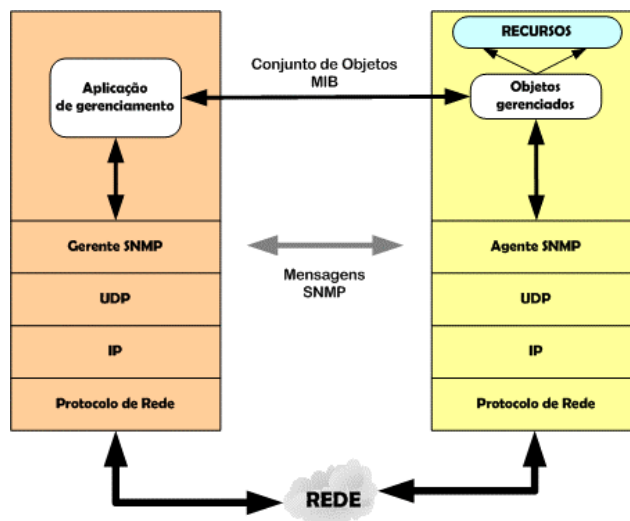


Figura 5 – Comunicação SNMP entre uma aplicação gerente e seu agente.

O uso dos protocolos UDP/IP referenciando o modelo OSI/ISO em operações SNMP, são fundamentais para a manutenção de algumas características operacionais do protocolo, tais como:

Roteamento : a camada de rede permite a escolha de uma nova rota em caso de uma falha, garantindo assim, que a gerência continue operando durante a mesma.

Independência do Meio de Transmissão: a camada de rede torna transparente a arquitetura física da rede.

Checksum: a camada de transporte controla o checksum das mensagens, garantindo sua integridade.

Multiplexação e Demultiplexação: os serviços da camada de transporte permitem o relacionamento n-para-n que o SNMP utiliza.

Fragmentação e Agrupamento: isto permite que pacotes SNMP não possuam tamanho fixo, sendo esta a característica fundamental deste protocolo.

Porém, a principal e mais importante característica é que os recursos necessários para um dispositivo gerenciado suportar UDP/IP são mínimos e a CPU é utilizada somente quando existe a necessidade de transmissão ou recepção de dados. Além disso, o *overhead* gerado por uma transação UDP/IP é muito baixo. Apesar do UDP requerer 28 octetos (sem a opção IP), o tráfego na rede é menor comparativamente a protocolos orientados a conexão.

Para o SNMP, o serviço de transporte sem conexão está especificado no protocolo, já que é prerrogativa básica deste continuar operando, mesmo se existirem problemas de performance na rede. Se o SNMP utilizasse um serviço de transporte orientado à conexão, este passaria a possuir a responsabilidade de garantir a confiabilidade dos dados transmitidos, o que exigiria a adição de algumas características ao protocolo, tais como:

- Estabelecer e finalizar conexões entre gerentes e agentes;
- Manter a conexão aberta para cada agente da rede;
- Manter um número fixo de conexões abertas.

Todas estas alternativas causariam diversos problemas indesejáveis: a primeira e a segunda requereriam um aumento de recursos de CPU, além do tráfego adicional gerado apenas para manter a conexão aberta. A terceira, implicaria que as estações

gerente e agente implementassem um algoritmo para aumentar o número de conexões, complicando a operação de ambos e provendo serviços desnecessários ao SNMP.

5.1. Conjunto de Funções SNMP

Todas as funções do gerente são variações de dois comandos, SET (alterar) ou GET (consultar), executados sobre uma MIB. Além das operações sobre a MIB, o protocolo implementa um comando para a detecção de eventos assíncronos, que podem ocorrer com os agentes na rede, denominado TRAP. Os principais comandos SNMP descritos na versão 1 do protocolo são:

getRequest : consulta um objeto MIB;
 getNextRequest : percorre a tabela MIB;
 setRequest : altera o valor de um objeto MIB;
 getReponse : resposta de um Request;
 trap : permite aos elementos da rede gerar eventos para as estações de gerenciamento. A RFC1157 (IAB, 1993) define cinco tipos de traps: coldStart, warmStart, link Down, linkUp, authenticationFailure, egpNeighborLoss e enterpriseSpecific.

A relação entre os elementos que compõe as funções de comunicação de gerentes e agentes SNMP em uma estrutura aplicada ao controle semafórico, pode ser vista na Figura 6.

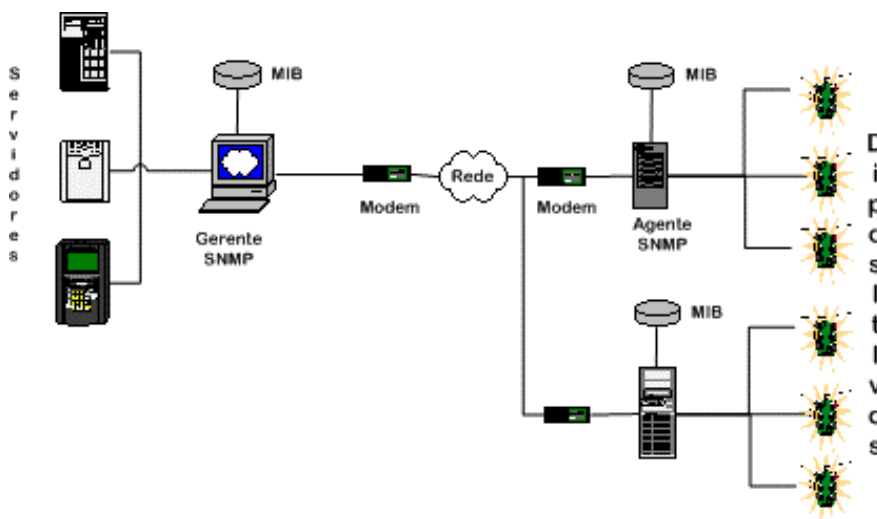


Figura 6 – Modelo de gerenciamento SNMP aplicado ao tráfego.

5.2. Agentes SNMP

Os agentes SNMP armazenam informações referentes ao estado dos objetos gerenciados em um banco de dados local . O conjunto de estados de um objeto representa a situação atual e histórica de um elemento gerenciado, que pode ser um computador, um roteador ou um dispositivo qualquer, que possua a capacidade de comunicação com o mundo externo. São os agentes que efetuam operações sobre as variáveis de estado, requisitando informações ou provocando mudanças nas ações dos dispositivos.

Os agentes são os protagonistas do processo de interação entre o mundo externo e os dispositivos gerenciados, porém não cabe a estes o comando das ações executadas. É o gerente SNMP que solicita dados ou determina uma ação através de requisições remotas, ficando o agente apenas com uma operação independente, a de reportar falhas ou alguns tipos de ocorrências geradas por seus dispositivos.

5.3. Gerentes SNMP

O controle efetivo dos dispositivos conectados a rede ocorre na estação de gerenciamento, onde um ou mais processos gerentes trocam informações com os processos agentes. As aplicações gerente executam o processamento das informações contidas na rede, interpretam resultado de consultas e definem as ações. Este tipo de gerência denominada mestre-escravo, permite que os agentes sejam projetados de modo muito simplificado, modelados apenas para executar as tarefas que lhe forem designadas, podendo assim, serem implementados em dispositivos com baixa capacidade de processamento.

As estações gerentes também servem como interfaces entre o operador humano e os componentes gerenciados, traduzindo o conjunto de informações geradas pelos dispositivos da rede para os usuários.

5.4. Pacotes de Informação SNMP

Sendo o SNMP implementado numa abordagem assíncrona, uma entidade não precisa aguardar a resposta após enviar uma mensagem, podendo continuar seu processamento ou enviar outra. Assim, uma interação SNMP consiste em uma solicitação, seguida de uma resposta (Rose, 1998). A consequência desta forma de operação é que os agentes devem implementar seus próprios controles para garantir o envio dos pacotes de dados, *Package Data Units -PDU* .

Um PDU, é um conjunto de bits codificados e agregados, que posteriormente é enviado sem partição de seu conteúdo ao seu destino. Após a construção do PDU o pacote é autenticado e os endereços de origem e destino são adicionados. Além disso, podem ser realizados serviços de

criptação ou inclusão de código de autenticação. Ao pacote resultante são adicionadas a versão e comunidade e então este é enviado para a camada inferior da pilha de protocolos.

A camada de transporte se encarrega de enviar a mensagem ao receptor. Já no destino, o processo é inverso, verifica-se existência de algum erro no pacote, então se tudo estiver correto, obtém-se a versão e somente após isso é que os dados serão processados e gerada uma mensagem de resposta.

Existem situações onde dispositivos da rede não reconhecem a pilha de protocolos necessários ao gerenciamento SNMP. Para garantir que estes dispositivos possam ser gerenciados o modelo SNMP define o conceito de *proxy*. Este agente *proxy* tem a função de converter mensagens SNMP em proprietárias e vice-versa, garantindo que estações de gerenciamento recolham informações ou indiquem ações sobre estes. A Figura 7 descreve o formato das mensagens SNMP.

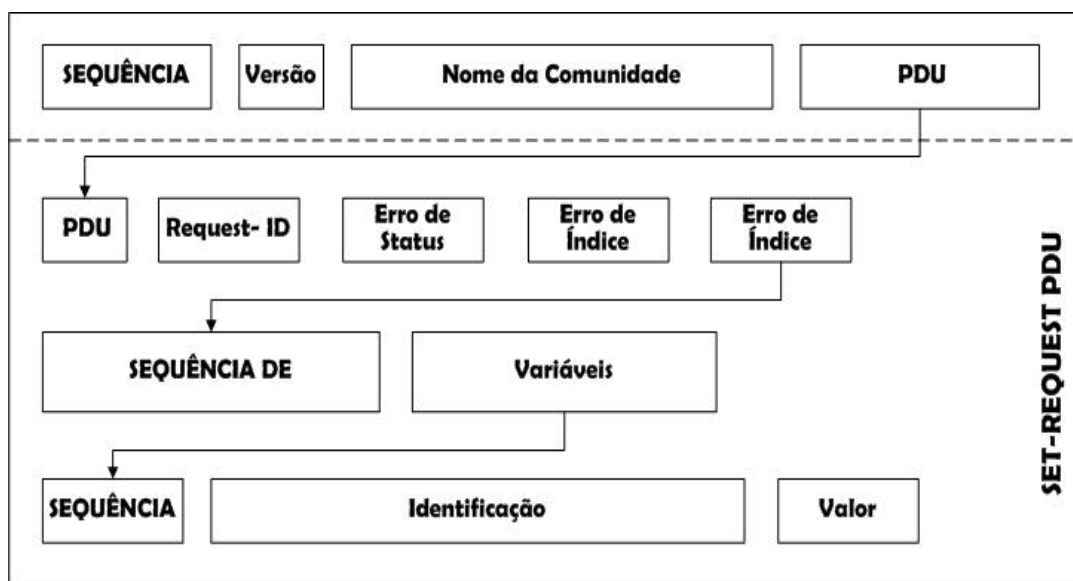


Figura 7 – Formatos de Mensagens SNMP (FHWA, 1999d).

Neste anexo estão descritos alguns dos principais padrões utilizados dentro da NIA, para atender as demandas de comunicação entre dispositivos SIT. Existem outros padrões tão importantes quanto os aqui citados, porém a escolha destes se deve a sua abrangência dentro da arquitetura. Mesmo que centrais de gerenciamento ou dispositivos venham a utilizar diferentes tecnologias de hardware ou software para a troca de dados, os conceitos anteriormente descritos como a ASN.1, MIB e SNMP estarão presentes, na descrição, armazenamento e acesso a estes dados.

Anexo 2

Simple Fixed Message Protocol - SFMP

Este anexo descreve as estruturas de pacotes de dados, comandos e campo de dados SFMP, além de um exemplo de montagem de um destes pacotes, de acordo com (FHWA, 2002b). Os agentes SFMP, devem ser capazes de receber e processar os seguintes comandos:

Set-Request-PDU
Get-Request-PDU
Set-Request-NoReplay-PDU

Além destes comandos, cada agente SFMP deve possuir a capacidade de responder as solicitações, emitir mensagens de erro e TRAPs, como descrito abaixo:

Set-Response-PDU
Get-Response-PDU
ErrorResponse-PDU
Trap-PDU

Como no SNMP e STMP a sintaxe para a especificação do protocolo é a ASN.1, a estrutura do pacote de dados é descrita da seguinte forma:

```
SFMP-Data-Packet ::= CHOICE {
    sfmp-get           [0] SFMP - GetRequest-PDU
    sfmp-set           [16] SFMP - SetRequest-PDU
    sfmp-set-no-reply [32] SFMP - SetRequest-NoReply -PDU
    sfmp-get-response [PRIVATE 0] SFMP- GetResponse-PDU
    sfmp-set-response [PRIVATE 16] SFMP- SetResponse-PDU
    sfmp-error         [PRIVATE 32] SFMP- ErrorResponse-PDU
    Sfmp-trap          [PRIVATE 48] SFMP-Trap-PDU
    ...
}
```

Cada um dos PDUs acima referenciado, baseia-se sobre o mesmo núcleo de estrutura de dados denominado, *Data Structure Fields*, descrito abaixo:

```
SFMP-PDU ::= SEQUENCE {
    Version           ENUMERATED{versão 1,...}   DEFAULT v1
    Community-name    OCTET STRING              DEFAULT public
    Request-number    INTEGER (0..255)          OPTIONAL
    Error-data        Error-Data                OPTIONAL
    Message-oid       RELATIVE-OID              OPTIONAL
    -- from {isso org dod internet private enterprises nema }
    Data              OBJECT-TYPE.&Syntax        OPTIONAL
    Trap-data         Trap-Data                  OPTIONAL
    ...
}
```

A seguir estão descritas as subestruturas dos pacotes de dados *Data-Packet* e dos campos (*Data Structure Fields*):

```

SFMP-GetRequest-PDU          ::= SFMP-PDU
SFMP-SetRequest-PDU          ::= SFMP-PDU
SFMP-SetRequest-NoReply-PDU  ::= SFMP-PDU
SFMP-GetResponse-PDU         ::= SFMP-PDU
SFMP-SetResponse-PDU         ::= SFMP-PDU
SFMP-ErrorResponse-PDU       ::= SFMP-PDU
    Error-Data ::= SEQUENCE {
        error-status      Error-Status,
        error-index       Error-Index
    }
SFMP-Trap-PDU ::= SFMP-PDU
    Trap-Data ::= SEQUENCE {
        eventLodID        INTEGER (1..65535),
        trap-time         Counter,
        trap-data         Opaque
    }

Version ::= ENUMERATE {version-1 (1), ...}
Request-Number ::= INTEGER (0..255)
Community-Name ::= OCTET STRING
Message-OID ::= RELATIVE-OID

```

O exemplo a seguir descreve a codificação usando OER e a montagem de um comando *get-request* no SFMP e seu respectivo *get-response* do objeto MIB, *globalTime.0*:

```

80          SFMP Get-Request Data-Packet
            CHOICE = [0]
            SFMP-GetRequest-PDU = SEQUENCE
14          Preâmbulo = 0001 0100 =
            Bit 1 = 0 – extension absent
            Bit 2 = 0 – default version = version 1
            Bit 3 = 0 – default community name = “public”
            Bit 4 = 1 – request-number present
            Bit 5 = 0 – error-data absent
            Bit 6 = 1 – message oid present
            Bit 7 = 0 – data absent
            Bit 8 = 0 – trap-data absent
01          Request number = 1
06 04 02 06 03 01 00 Data = globalTime.0 = 975463200=November 29,200
                        At 2:00 am UTC.

```

```

80          SFMP Get-Response Data-Packet
            CHOICE = [PRIVATE 0]
            SFMP-GetRequest-PDU = SEQUENCE
14          Preâmbulo = 0001 0010 =
            Bit 1 = 0 – extension absent
            Bit 2 = 0 – default version = version 1

```

Bit 3 = 0 – default community name = “public”

Bit 4 = 1 – request-number present

Bit 5 = 0 – error-data absent

Bit 6 = 0 – message oid present

Bit 7 = 1 – data present

Bit 8 = 0 – trap-data absent

01

3A 24 63 20

Request number = 1

Data = globalTime.0 = 975463200=November 29,200

At 2:00 am UTC.

Anexo 3

Simple Transportation Management Protocol - STMP

Este anexo descreve um exemplo do processo de montagem de um Objeto Dinâmico, partir de um comando STMP *set-request* de acordo com (FHWA, 2002b). Para a execução de um comando sobre um dispositivo SIT usando o STMP, são necessários que alguns passos sejam seguidos. No primeiro passo, o objeto dinâmico é definido, como no exemplo abaixo:

```
DynObjConfigOwner .3::= "Exemplo" {
    DynObjVariable.3.1::= globalTime.0
    DynObjVariable.3.2::= globaldayTimeSaving.0
    DynObjVariable.3.3::= controller-standardTimeZone.0
    DynObjVariable.3.4::= eventClassDescription.1.1
}
```

Os *dynObjVariable* 3.1, 3.2, 3.3 e 3.4, fazem parte do conjunto de objetos definidos sob o nó NEMA

O exemplo a seguir descreve a codificação usando OER e montagem de um STMP *set-request* do objeto dinâmico 3. O cabeçalho do objeto dinâmico é formado por um único byte dividido em duas partes de 4 bits cada, como segue abaixo:

Cabeçalho		Campo de informação	
1	Tipo da Mensagem	Identificação do Objeto	

BIT	CONTEÚDO	DESCRIÇÃO
7	FORMATO DO PDU	7
	0	Reservado para SNMP e algum uso futuro
	1	Indicado para pacotes STMP e SFMP
6-4	Tipo da Mensagem	
	000	STMP-GetRequest-PDU
	001	STMP-SetRequest-PDU
	010	STMP-SetRequest-NoReply-PDU
	100	STMP-GetNextRequest-PDU
	101	STMP-GetResponse-PDU
	110	STMP-SetResponse-PDU
	111	STMP-ErrorResponse-PDU
3-0	ID do Objeto	
	0000	Reservado para SFMP
	0001-1101	ID do STMP "objetos dinâmicos"
	1110	Reservado
	1111	Reservado

Desta maneira, a montagem em tempo de execução de um identificador de objeto dinâmico STMP *set-request* 3, pode ser descrita da seguinte forma:

Protocolo	Mensagem	ID Objeto	=	Cabeçalho
1	001	0011		10010011

STMP	SetRequest	Objeto "Exemplo"	93 (OER)
------	------------	------------------	----------

Seguindo o processo de codificação do objeto dinâmico, o restante dos objetos que fazem parte do pacote de dados a ser enviado ao agente devem ser codificados seguinte forma:

93		SFMP Get-Request Data-Packet
Information Field		
3A 24 63 20	Variable 3.1 =	globalTime.0 = 975463200=November 29,200 At 2:00 am UTC
03	Variable 3.2 =	globaldayTimeSaving.0 = enableUSDST
FF FF B9 B0	Variable 3.3 =	controller-standardTimeZone.0= 1800= EST
06 53 61 6D 70 6C 65	Variable 3.4 =	eventClassDescription.1 (6 bytes)="Exemplo"

Anexo 4

Request for Comments – RFC
relacionados ao SNMP.

Este anexo apresenta algumas das versões das *Request for Comments* - RFC e normas ISO, citadas na documentação distribuída pela NEMA e utilizada, como referência para a implementação do protocolo NTCIP.

- RFC 1212 – *Concise MIB (Management Information Base) for TCP/IP*
- RFC 1213 – *Management Information Base II*
- RFC 1215 – *A Convention for Defining Traps*
- RFC 1414 – *Identification MIB*
- RFC 1643 – *Ethernet-Like Interface Types MIB*
- ISO/IEC 8825-1 – *ASN.1 (ASN.1 Encoding Rules and Basic Encoding Rules - BER)*

Anexo 5

Parte do Código Fonte da Classe Codificar

```

void Config::Decodifica(AnsiString linha)
{
    AnsiString    tagVersao, tagComando, tagIndex, tagSequenceOfLength,
                 tagSequenceLength, tagOIDLength, tagEndereco,
                 tagTipo, tagValue, tagValueLength, tagStatus;

    int k,i,j=59;

    tagVersao=linha.SubString(9,2);
    tagComando=linha.SubString(27,2);
    tagStatus=linha.SubString(41,2);
    tagIndex=linha.SubString(47,2);
    tagSequenceOfLength=linha.SubString(51,2);

    while (i<((HexToInt(tagSequenceOfLength)*2)))
    {
        tagSequenceLength=linha.SubString(55+i,2);
        tagOIDLength=linha.SubString(j+i,2);
        j=j+2;
        tagEndereco=linha.SubString(j+i,HexToInt
        (tagOIDLength)*2);
        j=j+HexToInt(tagOIDLength)*2;
        tagTipo= linha.SubString(j+i,2);
        j=j+2;

        if (tagTipo=="02"){
            tagValueLength = linha.SubString(j+i,2);
            j=j+2;
            tagValue =linha.SubString(j+i,HexToInt(tagValueLength)*2);
            j=j+ HexToInt(tagValueLength)*2;
        }
        if (tagTipo=="1A")
            tagValueLength = linha.SubString(j+i,2);
            j=j+2;
            tagValue =
linha.SubString(j+i,HexToInt(tagValueLength));
            j=j+HexToInt(tagValueLength);
        }
        i=i+(HexToInt(tagSequenceLength)*2)+4;
        dadosDeco[k].comando=tagComando;
        dadosDeco[k].endereco=tagEndereco;
        dadosDeco[k].valor=tagValue;
        dadosDeco[k].tipo=tagTipo;
        k++;
    }
    dadosDeco[0].contador=k;
}

```

Anexo 6

Ambiente de Microsimulação de Tráfego
Fornecido pelo SITRA-B+

1. Descrição da Operação do Sitra-B+.

O software SITRA-B+ é estruturado de forma a oferecer ao usuário um ambiente de simulação de tráfego urbano baseado em diretrizes definidas por este usuário. Todos os parâmetros da simulação são descritos em arquivos texto. Cada um destes arquivos se refere à parte do cenário da simulação. Assim, são definidos os parâmetros que identificam as vias, sua largura, comprimento, que tipo de veículos e com que frequência surgirão no sistema, velocidades, etc.

Uma vez definido o cenário, o software executa a simulação, obedecendo dois critérios: no primeiro além de criar o fluxo de veículos o software opera sob um regime semafórico pré-definido. Em um segundo critério, o SITRA-B+ aceita que os estados semafóricos sejam alterados segundo os critérios de um operador externo. É sob esse segundo cenário que a aplicação NTCIP aqui desenvolvida opera. A figura abaixo descreve um dos arquivos de entrada do SITRA-B+, utilizado para configurar quais as conversões permitidas em cada cruzamento.

04140401	14000400	0414	0401	LINK	ALL	50	65	1800
05010514	05001400	0501	0514	LINK	ALL	50	65	1800
06140601	14000600	0614	0601	LINK	ALL	50	65	1800
07010714	07001400	0701	0714	LINK	ALL	50	65	1800
08140801	14000800	0814	0801	LINK	ALL	50	65	1800
09010913	09001300	0901	0913	LINK	ALL	50	65	1800
10131001	13001000	1013	1001	LINK	ALL	50	65	1800
11011113	11001300	1101	1113	LINK	ALL	50	65	1800
12131201	13001200	1213	1201	LINK	ALL	50	65	1800
13011401	13001400	1301	1401	LINK	ALL	50	65	1800
14021302	14001300	1402	1302	LINK	ALL	50	65	1800
01131013	1300	0113	1013	INTERSECTION	ALL	50	1800	

Uma vez que todos componentes da simulação foram definidos, sua execução ocorre de forma visual e os dados sobre o seu estado são disponibilizados, também em arquivos texto. Durante a simulação, as informações de saída são acessíveis somente no final de cada “passo de simulação”. Um “passo de simulação” é um intervalo de tempo Δ no qual ocorreram X simulações, por exemplo, um segundo de tempo real pode representar uma hora de tráfego no cenário. É somente após um “passo de simulação”, que são liberadas as informações sobre o estado das vias e seus componentes. Estes dados são sempre relativos aos veículos na via, tais como, ocupação, tipo de veículo, etc. Além desses dados, no final da simulação outros arquivos texto que permaneceram vazios durante todo o processo, são disponibilizados com uma gama variada de dados estatísticos sobre a mesma. A figura a seguir mostra um dos arquivos de saída disponibilizados a cada “passo de simulação”, no qual estão dados referentes a ocupação da via no tempo Δ de simulação.

1001300	1	0.000000	0	0
13000200	0	0.000000	0	0
3001400	2	0.000000	0	0
14000400	0	0.000000	0	0
5001400	1	0.000000	0	0
14000600	2	0.000000	0	0
7001400	5	0.000000	0	0
14000800	0	0.000000	0	0
9001300	6	0.000000	0	0
13001000	1	0.000000	0	0

A interface gráfica disponível no SITRA-B+ permite a criação de vias simuladas com grande riqueza de detalhes. Já que, além da modelagem da via com suas pistas, interseções, dispositivos de tráfego entre outros, permite a inserção de um fundo gráfico. Este fundo gráfico pode ser formado por imagens digitalizadas de uma via real ou uma criação do projetista do ambiente. Na figura abaixo é apresentada uma dessas interfaces.

