Universidade Federal do Ceará
Departamento de Computação
Mestrado em Ciências da Computação

Dissertação de Mestrado

SGME, Um Sistema Genérico de Monitoração de Redes de Computadores Dirigido a Eventos

por

Aecio Paiva Braga

Orientador: Prof. Dr. José Neuman de Souza
Co-Orientador: Prof. Dr. Javam de Castro Machado

SGME, Um Sistema Genérico de Monitoração de Redes de Computadores Dirigido a Eventos

Aecio Paiva Braga

Dissertação apresentada ao curso de Mestrado em Ciência da Computação da Universidade Federal do Ceará, como parte dos requisitos para a obtenção do Grau de Mestre em Ciência da Computação.

Composição da Banca Examinadora:

_____

Prof. Dr. José Neuman de Souza(DC/UFC)

_____

Prof. Dr. Javam de Castro Machado(DC/UFC)

_____

Prof. Dr. Djamel H. Sadok(CI/UFC)

Aprovada em 11 de Março de 2003

Universidade Federal do Ceará
Centro de Ciências
Departamento de Computação
Mestrado em Ciências da Computação

Aecio Paiva Braga

SGME, Um Sistema Genérico de Monitoração de Redes de Computadores Dirigido a Eventos

Trabalho apresentado à Pós-Graduação em Ciências da Computação do Centro de Ciências da Universidade Federal do Ceará como requisito parcial para obtenção de grau de Mestre em Ciências da Computação.

Orientador: Prof. Dr. José Neuman de Souza
Co-Orientador: Prof. Dr. Javam de Castro Machado

Agradecimentos

Primeiramente, gostaria de agradecer ao meu orientador Professor José Neuman de Souza e ao Diretor do Núcleo de Processamento de Dados(1999-2003), Professor Mauro Cavalcante Pequeno, cujas recomendações me abriram as portas do Curso de Mestrado em Ciências da Computação da Universidade Federal do Ceará.

Aos meus amigos e familiares, pelo apoio que sempre manifestaram no decorrer do curso. Em especial, ao Hermes, Liege, Alberto e Belo, pelas várias revisões dos meus artigos em português e inglês. Sem palavras para expressar a minha gratidão.

A todos os professores e funcionários do Departamento pela contribuição e apoio.

Pedro e Socorro Braga. Meus pais e eternos exemplos de luta e superação de dificuldades, vocês foram o início de tudo.

Esse trabalho é totalmente dedicado à Sandra Maria Coelho Rodrigues, minha companheira e amiga de todas as horas.

Resumo

No campo da monitoração das redes de computadores, o SGME, é um *framework* de suporte ao desenvolvimento, apoiado pela modelagem automática de dados e eventos, de programas dirigidos para o gerenciamento de domínios que podem surgir no cenário das redes monitoradas por meio do protocolo SNMP. Esse *framework* automatiza o agrupamento de objetos das MIB's dos diversos Agentes SNMP, produzindo estruturas de dados que representam Domínios e Eventos. Essas estruturas são, automaticamente, transformadas em Tabelas e em Visões de um Banco de Dados Relacional. As Visões implementam os mecanismos de detecção os Eventos definidos no *framework*. O SGME também disponibiliza um Agente Monitorador que, automaticamente, reconhece, coleta e armazena as informações dos Domínios.

Abstract

In the area of computer network monitoring, the SGME is a development framework, supported by automatic modeling of data and events, of programs driven to management domains in the scenery of the SNMP-based network monitoring. That framework automates the grouping of MIB's objects of the several SNMP Agents and produces data structures that represent Domains and Events. Those structures, automatically, becomes relational database's tables and visions. The visions implement the perception mechanisms of Events defined in the framework. SGME also provides an Monitoring Agent that, automatically, recognizes, collects and stores the Domains' information in the database.

Sumário

Lista de Figuras

Lista de Tabelas

Capítulo 1 - Introdução

1.1 Contextualização

Tipicamente, nos primeiros momentos da existência de uma rede local de computadores, ela é relativamente pequena e totalmente homogênea, conseqüentemente o seu gerenciamento é, nesses momentos, em geral, desembaraçado e pode ser feito por meio da ação direta de um operador especializado humano.

Com o passar do tempo, a rede cresce, tendendo a ficar heterogênea e a suportar uma quantidade sempre crescente de serviços e usuários, tornando-se cada vez mais vital para os negócios das organizações. Desse modo, o consumo de tempo e o volume de dados envolvidos no seu gerenciamento, também, ficam cada vez maiores.

Diante do crescimento do tamanho da rede, do aumento de consumo de tempo e do volume de dados, a atividade de gerência exercida diretamente pelo operador humano começa a tornar-se mais complexa. Com isso, surge um ambiente que demanda um sistema de automação do processo de gerenciamento da rede. Esse sistema de gerência depende de dois fatores. O primeiro fator refere-se à monitoração dos estados dos nodos da rede. Por meio da monitoração, busca-se reconhecer e correlacionar os eventos que ocorrem na rede. Já o segundo fator refere-se aos conhecimentos e à capacidade dos especialistas em redes de selecionar e correlacionar estados.

O crescimento da rede, normalmente, envolve a utilização de equipamentos de diferentes marcas. Há mais de uma década, a grande maioria dos computadores utiliza o protocolo TCP/IP que foi projetado para resolver o problema da interoperabilidade e compartilhamento de recursos entre os equipamentos produzidos por diferentes fabricantes. Diante dessa

heterogeneidade, surgiu a necessidade da padronização de uma ferramenta que apoiasse a gerência das redes formadas com esses tipos de equipamentos. Para isso, foi desenvolvido o protocolo *Simple Network Management Protocol* - SNMP.

O SNMP foi desenvolvido para prover aos diversos fabricantes de equipamentos uma ferramenta de suporte às aplicações de gerenciamento de redes interoperáveis [Stallings1999]. Ele é, atualmente, o protocolo mais implementado em todo o mundo, sendo suportado por um número constantemente crescente de equipamentos de rede e é projetado para fazer exatamente o que o seu nome sugere: apoiar a execução, de um modo relativamente simples, do gerenciamento dos recursos de hardware e software de uma rede.

1.2 Motivação

Os vários trabalhos na área de gerenciamento de redes, [LI 2000] [LO 1998] [HO 2000] [BURGESS 2000], em geral, buscam identificar e perceber Eventos, como meio para detecção, diagnóstico e correção de anomalias das redes de computadores. Essas abordagens utilizam, por exemplo, cálculos probabilísticos, grafos e raciocínios baseados em regras e em casos. Em geral, essas aplicações são alimentadas por alarmes ou arquivos de *logs*. Elas buscam, a partir dos eventos, antever tendências e anomalias, detectar, isolar e corrigir as causas originais dos eventos e mapear as observações físicas com os estados dos objetos gerenciados da rede.

No âmbito da gerência apoiada pelo protocolo SNMP, os sistemas são alimentados por informações oriundas das *Management Information Bases* - MIBs dos Agentes e/ou das sondas de Monitoração de Redes Remotas (*Remote Network Monitoring* - RMON).

Uma MIB é um conjunto de informações dinâmicas cujos valores mudam em função do tempo e das suas localizações. Portanto, é necessário que os objetos dos diversos Agentes SNMP e/ou sondas RMON sejam agrupadas, com base em

supostos relacionamentos, para que um Domínio de Gerenciamento possa ser obtido em conformidade com a definição de [PERKINS 1997]. Além disso, as MIBs RMON e RMON2 representam processos geradores de estatísticas, de eventos e traps que, eventualmente, podem se tornar um *overhead* na capacidade de computação das sondas.

Atualmente, a quantidade e os tipos de aplicações de gerenciamento de redes é grande e suas características computacionais variam bastante. Infelizmente, o desenvolvimento dessas aplicações não leva em conta as necessidades e os domínios de gerenciamento em geral [Hariri 2000]. Isso induz a necessidade de um mecanismo, genérico e automático, de modelagem capaz de representar tais domínios de modo a facilitar a percepção de eventos já que esse tipo de mecanismo não está contemplado nas especificações do protocolo SNMP. Ao mesmo tempo, a implementação fora das sondas, dos domínios modelados, pode torná-las computacionalmente mais ágeis.

A automação do processo de gerência da rede necessita, no mínimo, de um sistema de monitoração que permita, de modo simplificado, ao administrador definir diversos domínios de gerenciamento da rede. Essas definições conduzem, com bastante precisão, o foco das atenções do administrador para os estados dos nodos da rede, tendo em vista a grande quantidade de informações gerenciais que se pode monitorar simultaneamente.

A origem dos estados monitorados encontra-se nas MIBs. Elas são extensas e dispõem apenas de visões instantâneas dos valores das suas variáveis. Isso indica que a coleta e a análise dos dados das MIBs tornam-se bastante laboriosas diante do grande número de variáveis e da constante variação do estado geral aparente de um nodo da rede. Essa complexidade aumenta diante das diversas possibilidades de correlacionamento dos estados dos diversos nodos.

1.3 Objetivos do trabalho

Diante das constatações apresentadas anteriormente, esse trabalho tem como objetivo geral, no campo da monitoração, a produção de uma plataforma de suporte ao desenvolvimento, apoiado pela modelagem automática de dados e eventos, de aplicações de gerenciamento dos domínios que podem surgir no cenário das redes de computadores monitoradas por meio do protocolo SNMP.

Essa plataforma deve permitir o agrupamento das informações oriundas dos Agentes SNMP e/ou das sondas RMON e a criação de um repositório para essas informações em um banco de dados, de modo transparente. Atributos para registro de data e hora devem ser incluídos nesses modelos de dados, indicando os momentos de requisição e de recepção das informações dos domínios modelados. Eventos serão representados por Regras formuladas a partir dos atributos dos repositórios das informações monitoradas. Um Agente Monitorador proverá transparência ao processo de monitoração das informações dos domínios de gerenciamento.

Dessa forma, as aplicações podem ficar dedicadas, exclusivamente, aos procedimentos gerenciais, isentam-se dos *pollings*, e decisões podem ser tomadas a partir da detecção dos eventos implementados como visões do banco de dados ou a partir de outras análises feitas sobre as informações dos repositórios. Assim, várias pequenas aplicações podem, cada uma, gerenciar seu próprio domínio, facilitando o desenvolvimento de um Sistema de Gerenciamento.

O repositório dos dados coletados, isto é, a tabela que contém as informações de um Domínio de Gerenciamento, pode, eventualmente, ser capaz de alimentar processos de *datamining*, *datawarehouse*, estatísticos, etc.

Como objetivo específico, a plataforma deve constituir um *framework* que deve permitir a construção de sistemas de gerenciamento de redes de computadores, provendo:

- interface operacional *Web*;
- facilidades para definição de domínios de gerenciamento;
- abstração do processo de coleta e armazenamento de dados;
- abstração do protocolo SNMP nas fases não reativas do processo de gerenciamento(comunicação, estrutura de dados, etc.);
- facilidades para automatização da coleta, datação e armazenamento de informações;
- facilidades para automatização da detecção de Eventos.

1.4 Estrutura do trabalho

O capítulo 2 apresenta, de modo geral, a arquitetura, os elementos e funcionamento e um pequeno histórico evolutivo do protocolo SNMP. Apresenta, também, de modo sucinto, as abordagens para Monitoração das Redes Remotas no contexto SNMP.

O capítulo 3 dá uma visão das abordagens de alguns trabalhos na área de gerenciamento de redes baseados em Eventos, apresenta a motivação e as tecnologias que influenciam esse trabalho.

O capítulo 4 descreve os objetivos, a arquitetura e os componentes de um Sistema Genérico de Monitoração de Redes de Computadores Dirigido a Eventos - SGME.

O capítulo 5 apresenta o modelo do banco de dados que dá suporte ao desenvolvimento do SGME.

O capítulo 6 define a implementação e as funcionalidades do SGME e o capítulo 7 faz a conclusão dessa dissertação.

Capítulo 2 - O Protocolo SNMP

2.1 Introdução

As redes e os sistemas de processamento distribuído, atualmente, estão ficando cada vez mais vitais para o mundo dos negócios. Dentro das organizações, a tendência é que essas instalações se tornem maiores e mais complexas, suportando um número maior de aplicações e usuários. À medida que as redes crescem, dois fatores se mostram bem claros [Stallings1999]:

- As redes e as aplicações distribuídas estão, gradativamente, constituindo um ambiente indispensável para as empresas;
- O mau funcionamento desse ambiente pode afetar a qualidade dos serviços que as redes e as aplicações devem prover.

Ainda segundo Stallings, as grandes instalações não podem ser gerenciadas apenas com o esforço humano. A sua complexidade impõe a utilização de ferramentas que proporcionem a automatização do seu gerenciamento. A urgência da necessidade dessas ferramentas cresce e, quando a rede é formada por equipamentos de fabricantes diversos, a dificuldade de supri-las também aumenta. Nesse caso, ao passo em que as instalações crescem, ficando mais complexas e heterogêneas, ferramentas padronizadas são necessárias para que se possa gerenciar a diversidade de equipamentos da rede. Para isso, foi desenvolvido o protocolo SNMP.

2.2 O *Simple Network Management Protocol*

O *Simple Network Management Protocol - SNMP* foi desenvolvido para prover aos diversos fabricantes de equipamentos uma ferramenta de suporte às aplicações de gerenciamento de redes interoperáveis [Stallings1999].

O SNMP é o protocolo de suporte às aplicações de gerenciamento de redes atualmente mais implementado em todo o mundo. Ele é suportado por um número

crescente de equipamentos de rede, e é projetado para permitir a execução, com relativa simplicidade, do gerenciamento dos recursos de hardware e software da rede. Inicialmente, esse protocolo foi proposto para dar suporte ao gerenciamento de computadores interligados por meio do protocolo TCP/IP. Porém, a abordagem do gerenciamento baseado no protocolo SNMP é genérica o bastante para ser utilizada na gerência de vários tipos de sistemas, com por exemplo: redes de malhas viárias, redes de censores de temperaturas, redes de irrigação, etc. [Orfali1996].

O protocolo SNMP possui uma arquitetura de computação distribuída que consiste de quatro elementos básicos [Perkins1997]:

- Um ou mais nodos gerenciados (PCs, mainframes, roteadores, hubs, switchs, etc), cada um contendo uma entidade de processamento, chamada Agente;
- Pelo menos uma estação de gerenciamento contendo uma ou mais entidades de processamento chamadas de Aplicações Gerentes ou simplesmente Gerentes;
- Uma base de informações de gerenciamento em cada Agente, que descreve estados e estatísticas e que também controla a operação dos nodos da rede;
- O protocolo, propriamente dito, que é seguido pelos Agentes e Gerentes para trocas de mensagens de gerenciamento em conformidade com o paradigma Gerente/Agente.

2.2.1 O Paradigma Gerente/Agente

O Paradigma Gerente/Agente estabelece um modelo hierárquico no qual um Gerente troca mensagens com um Agente, enviando requisições e recebendo respostas. Ambos, Gerente e Agente, são aplicativos que provêem, essencialmente, funções e serviços gerenciais. Um Agente provê o acesso e a manipulação das informações de gerenciamento que podem ou não resultar em ações que repercutem diretamente na operação dos objetos, recursos de

hardware e *software*, de um nodo gerenciado. Ele também pode ter sob sua tutela um subagente que é responsável pelo seu próprio subconjunto de objetos gerenciados. No caso de um nodo ser operável por meio de um protocolo desconhecido pelo Gerente, então faz-se necessária a presença de um agente Proxy para compatibilizar a comunicação entre esse nodo e o Gerente [Ghetie1997].

A arquitetura descrita por Perkins(1997) e o esquema geral do funcionamento do protocolo SNMPv1 estão representados na Figura 2.1.

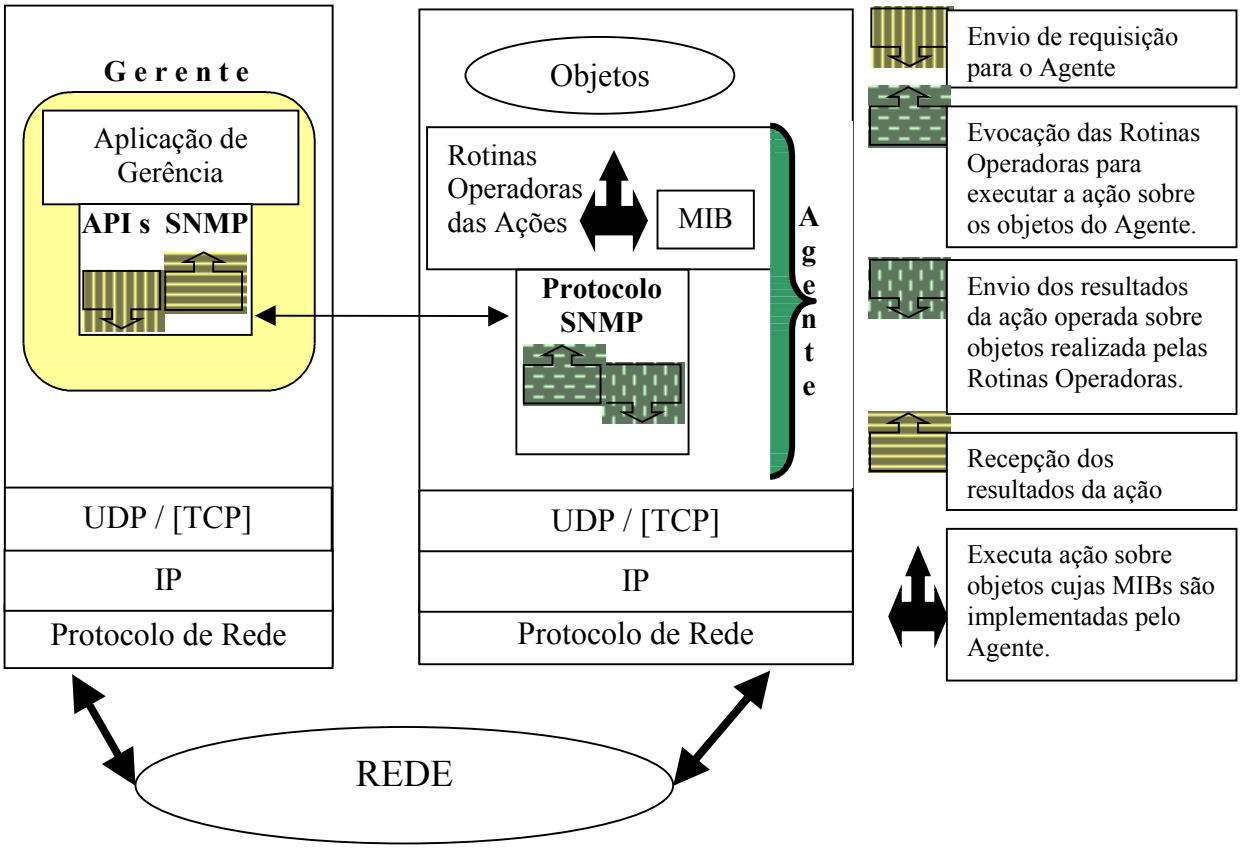

Figura 2.1 - Esquema de Funcionamento do Protocolo SNMP

2.2.1.1 O Gerente SNMP

Um Gerente SNMP é uma aplicação que intermedeia a interação entre o elemento humano e os nodos gerenciados que contêm um Agente. Essa aplicação provê as interfaces que permitem a um Administrador monitorar e controlar esses nodos para manter a qualidade dos serviços da rede. Isso é feito por meio do envio de mensagens que:

- Solicitam informações gerenciais de um nodo;
- Atribuem valores a essas informações de modo a interferir, de alguma maneira, no funcionamento do nodo.

2.2.1.2 O Agente SNMP

O Agente é um aplicativo que é executado por um nodo gerenciado da rede. Ele é o resultado da tradução, tanto das especificações contidas nos documentos designados como *Request For Comments* - RFC(RFC 1155,1156 e 1157) que definem todos os elementos do protocolo SNMP como das RFCs das diversas MIBs até então já produzidas, em um programa ou código executável de computador. O desenvolvimento e a utilização desse aplicativo, também, é chamado de instrumentalização de um nodo da rede. Essa instrumentalização implementa a Base de Informações Gerenciais ou MIBs, processa e responde as mensagens oriundas dos Gerentes e também emite, assincronamente, informações previamente categorizadas como importantes (*Traps*) para o(s) Gerente(s). Em última instância, é a implementação do protocolo SNMP propriamente dito.

2.2.2 A Base de Informações Gerenciais

A Base de Informações Gerenciais (*Management Information Base* - MIB) é um conjunto de especificações expressas por meio da Notação para Sintaxe Abstrata-1 (*Abstract Syntax Notation One* - ASN.1). Essas especificações descrevem as estruturas das informações gerenciais através do esquema chamado *Structure of Management Information* - SMI. Tais informações

representam objetos que constituem uma visão lógica dos recursos de hardware e software dos nodos da rede, os quais podem ser manipulados diretamente por um Agente. Tais recursos vão desde o meio de transmissão até detalhes do vários protocolos desses nodos [Stallings1999]. Os objetos das MIBs podem ser considerados como uma coleção de variáveis, em conformidade com as suas especificações SMI, cujos valores podem ser manipulados por um Agente por requisição de um Gerente.

Inicialmente, o protocolo SNMP padronizou uma base de informações gerenciais chamada MIB-I cujas especificações, posteriormente, foram ampliadas e passaram a compor um novo padrão, atualmente bastante difundido, chamado MIB-II.

As MIBs são formadas por objetos que caracterizam o nodo possuidor de um Agente. Esses objetos são identificados por uma seqüência de números inteiros, também chamada de Identificador do Objeto(*Object Identifier* - OID). Além do OID, uma macrotipagem é utilizada para definir um objeto. Ela é normalmente composta por:

- um nome;
- um tipo (*integer, counter, string, gauge*, ou *address* dentre outros);
- uma restrição de tamanho (quantidade de octetos que pode ocupar);
- um tipo de acesso ao objeto (*read, read/write, write, none*);
- um lista de valores que o objeto pode assumir e
- uma descrição textual do objeto.

A seqüência de números inteiros que forma um OID é na verdade o mapeamento de uma árvore hierárquica, Figura 2.2, na qual suas folhas são abstrações dos recursos dos nodos gerenciados.

A **posição** do inteiro identifica o **nível** (1 a n) em que aquele número se encontra na hierarquia da árvore. Já o **valor** do inteiro corresponde ao número de ordem da **ramificação** dentro de determinado nível.

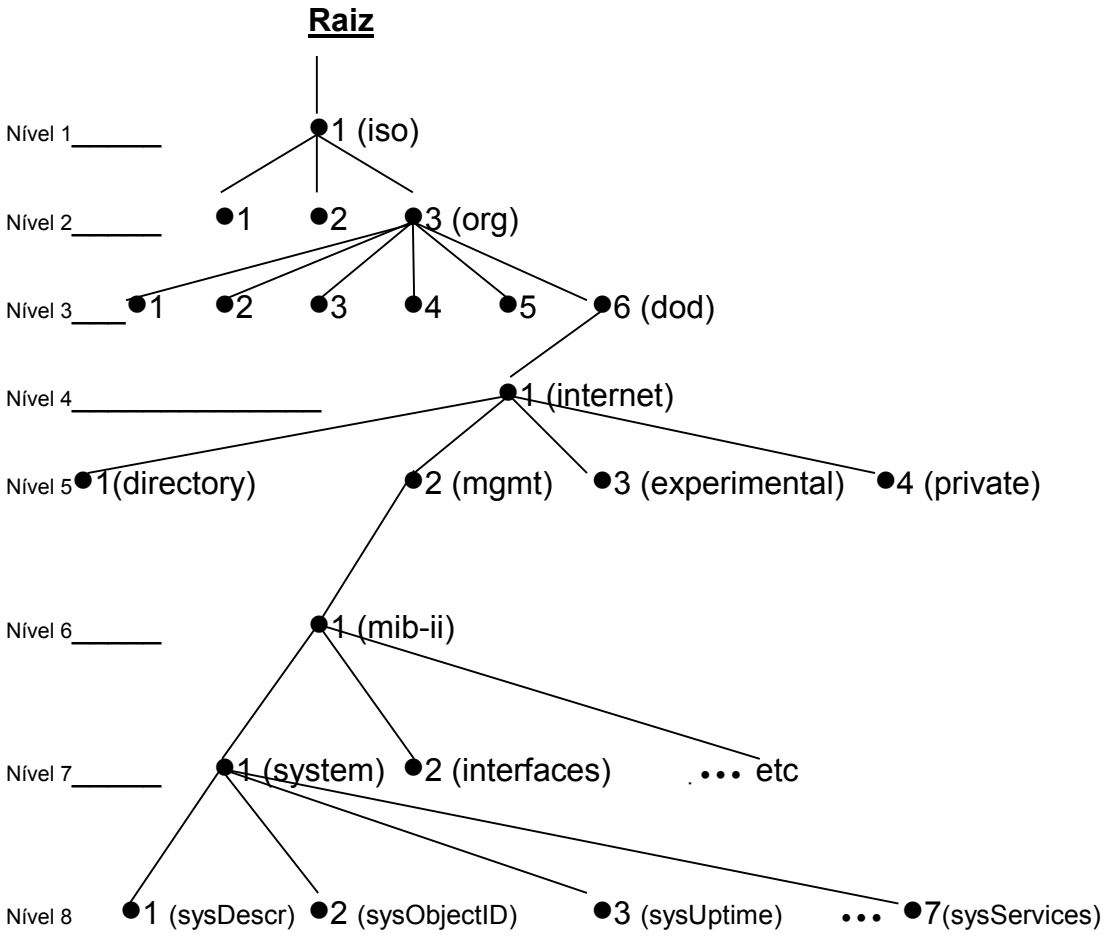

Figura 2.2 - Árvore Hierárquica

A tabela 2.1 exemplifica as definições dos objetos apresentados na Figura 2.2 pertencentes à MIB-II.

Tabela 2.1 - Folhas da Árvore Hierárquica

OID	Nome do Objeto	Valor / Macrotipagem
.1.3.6.1.2.1.1.1.0	iso.org.dod.internet.mgmt. mib-2.system.sysDescr	Hardware: x86 Family 6 Model 1 Stepping 7 AT/AT COMPATIBLE / Octet String / 256 / Read
.1.3.6.1.2.1.1.2.0	iso.org.dod.internet.mgmt. mib-2.system.sysObjectID	.1.3.6.1.4.1.311.1.1.3.1.3 / ObjectIdentifier / Read
.1.3.6.1.2.1.1.3.0	iso.org.dod.internet.mgmt. mib-2.system.sysUptime	3628484 / TimeTicks / Read
.1.3.6.1.2.1.1.7.0	iso.org.dod.internet.mgmt.	77 / Integer / Read / [0..127]

OID	Nome do Objeto	Valor / Macrotipagem
	mib-2.system.sysServices	

Analisando a tabela 2.1, é possível ver que o quarto objeto é identificado pelo OID <.1.3.6.1.2.1.1.7.0> onde:

- A primeira posição da seqüência indica nível 1(um) da árvore nascida da raiz, a segunda posição o nível 2(dois) e assim sucessivamente;
- O inteiro 1 da 1a. posição da seqüência indica a primeira ramificação do nível 1 da árvore a partir da raiz, o inteiro 3 da 2a. posição indica a 3a. ramificação do nível 2 da árvore, e assim sucessivamente;
- Na extremidade de cada ramificação da árvore, tem um nó que possui um nome. O encadeamento dos nomes dos nós desde o 1o. nível da árvore até a folha de um dado caminho compõe o nome do objeto gerenciado. No exemplo, forma-se a seguinte seqüência: *iso.org.dod.internet.mgmt.mib-2.system.sysServices*.
- O valor do objeto é 77 sendo do tipo *integer*. Ele só pode ser acessado para leitura e pode assumir valores entre 0 e 127.

Os objetos que compõem uma MIB podem ter o formato escalar ou colunar. Um objeto escalar é como uma variável simples, e um objeto colunar é como uma variável contida em uma tabela, quer dizer, ela só pode ser referenciada com o auxílio dos seus indexadores. Nesse sentido, o protocolo SNMP convenciona que a identificação de um objeto colunar é obtida pela concatenação do seu OID com os seus índices. E para manter a consistência com essa convenção, a identificação de uma instância de um objeto escalar é dada pelo seu OID concatenado com <.0>. Dessa forma, a construção de OIDs torna-se simples e uniforme.

A MIB-II está organizada em dez grupos funcionais de informações gerenciais, conforme apresentados na tabela 2.2.

Tabela 2.2 - Grupos de Informações da MIB-II [RFC 1213]

Grupo	Descrição
system	Especifica informações gerais sobre o nodo gerenciado, tais como descrição, local, nome do administrador, etc.
interfaces	Especifica informações sobre o tipo das interfaces físicas do nodo gerenciado, incluindo a contagem dos fluxos de dados em cada uma das suas interfaces.
at (*address translation*)	Especifica informações que permitem traduzir um endereço de protocolo de rede, tipicamente um endereço IP, para o seu endereço físico MAC (*Medium Access Control*). Esse grupo está obsoleto mas é mantido para garantir a compatibilidade com a MIB-I. A tradução de endereços, também, é promovida pelos grupos *ip, icmp, tcp, udp* e *egp*.
ip (*internet protocol*)	Contém informações relevantes para a configuração e operação do nodo. Também especifica as estatísticas sobre o tráfego que transita através do protocolo IP executado pelo nodo gerenciado.
icmp (*internet control message protocol*)	Especifica as estatísticas sobre os vários tipos de mensagens ICMP recebidas e enviadas pelo nodo gerenciado.
tcp (*transmission control protocol*)	Contém informações sobre o estado geral do funcionamento do protocolo TCP e especifica as estatísticas sobre tráfego que atravessam o protocolo TCP do nodo gerenciado.
udp (*user datagram protocol*)	Contém informações sobre estado geral do protocolo do UDP executado pelo equipamento da rede. Esse grupo também especifica as estatísticas sobre tráfego que transita através do protocolo UDP.
egp (*external gateway*	Contém informações sobre o estado geral do funcionamento do protocolo EGP, especifica as estatísticas tanto sobre tráfego que

Grupo	Descrição
protocol)	transita através do protocolo EGP bem como sobre os roteadores vizinhos, além de outros indicadores de funcionamento da vizinhança do nodo gerenciado.
transmisssion (*dot3*)	Contém contadores que detalham o funcionamento do meio de transmissão acoplado às interfaces do nodo gerenciado. Esse não é um grupo e sim um nó da hierarquia da MIB-II. Sob esse nó estão vários grupos, cada um específico para um tipo de interface de rede. Dentre eles, pode-se citar o grupo *dot3* referido como MIB EtherLink. O grupo *dot3*, propriamente dito, especifica as estatísticas sobre as operações do protocolo Ethernet que opera sobre os tipos de cabeamento mais usuais.
snmp (simple network management protocol)	O grupo especifica as estatísticas que detalha o funcionamento do próprio protocolo SNMP do nodo gerenciado.

2.2.3 O Protocolo

O protocolo SNMP, propriamente dito, estabelece os padrões para o intercâmbio de mensagens com os Agentes. Esse intercâmbio é feito por meio do *User Datagram Protocol* - UDP da série de protocolos TCP/IP.

Basicamente, esse protocolo padroniza os tipos e as funcionalidades das mensagens intercambiadas e um procedimento de autenticação dos Gerentes. Baseadas nessa padronização, hoje existem APIs(*Application Programming Interface* ou Interface de Programação de Aplicativos) que facilitam a montagem e o intercâmbio dessas mensagens. Com isso, a instrumentalização, isto é, a implementação das computações e das manipulações dos objetos, das MIBs ficam a critério dos desenvolvedores.

As funções dos tipos mais básicos de mensagens que o protocolo SNMP padroniza estão descritos na tabela 2.3.

Tabela 2.3 - Tipos de Mensagens SNMP [Perkins1997]

Tipo	Descrição	Origem / Destino
GET	Requisita um ou mais valores dos objetos das MIBs.	Gerente / Agente
GETNEXT	Requisita o valor de um objeto cujo OID está, na árvore da MIB, lexicográfica e imediatamente após o OID passado como argumento.	Gerente / Agente
SET	Requisita a atribuição de valores a um ou mais objetos das MIBs.	Gerente / Agente
TRAP	Reporta um evento ocorrido em um nodo gerenciado.	Agente / Gerente

De acordo com Stallings(1999), o intercâmbio de mensagens entre Gerente e Agente é iniciada com um processo de autenticação para evitar que um Agente atenda solicitações de algum Gerente não credenciado, ou atenda a solicitações não autorizadas de um Gerente credenciado. Para isso, o Agente possui uma tabela de credenciamento que contém as seguintes informações:

- Endereços IP dos Gerentes credenciados;
- Nomes das comunidades das quais fazem parte tanto o Gerente como o Agente. Um nome de comunidade funciona como uma "senha" do Gerente junto ao Agente;
- A ação que um Gerente associado a uma certa comunidade pode requerer ao Agente. Por exemplo: um Gerente pode fazer apenas requisições, enquanto outro pode fazer requisições e atribuições de valores aos objetos de uma MIB de um Agente.

Uma vez certificada a sessão, a troca de mensagens ocorre da seguinte maneira:

A) No lado do Gerente.

Uma aplicação de gerência envia mensagens, opcionalmente com auxílio de APIs SNMP, a um ou mais Agentes, requisitando a execução de uma ação sobre um ou mais objetos desses Agentes. Essas ações são requeridas por mensagens dos seguintes tipos:

1. *GET*: Mensagem que requisita a leitura dos valores dos objetos da MIB de um Agente SNMP, por exemplo, o tipo e o estado operacional de uma certa porta de comunicação desse mesmo nodo;
2. *SET*: Mensagem que requisita a atribuição de valores aos objetos da MIB de um Agente SNMP, por exemplo, o estado operacional de uma certa porta de comunicação pode ser alterado para "recarregando", forçando com isso a sua reinicialização.

Para ambos tipos de mensagens, ocorre, de modo transparente e síncrono, uma ação de recebimento de resposta na aplicação Gerente. Trata-se da devolução da mensagem recebida, com os mesmos OIDs originais, mas acompanhados dos valores resultantes da ação solicitada.

B) No lado do Agente.

O Agente funciona da seguinte maneira:

- Continuamente, verifica se alguma *Trap* foi acionada. Nesse caso, notifica a ocorrência de um evento significativo ao Gerente responsável pelo seu tratamento. Esta ação independe de quaisquer solicitações do Gerente;
- Paralelamente:
    1. O Agente recebe mensagens do Gerente;
    2. Verifica a existência, na sua MIB, dos objetos cujos OIDs são transportados pela mensagem recebida;
    3. Executa a ação(*Get/GetNext/Set*) solicitada sobre os objetos especificados, se possível;
    4. Envia uma mensagem de resposta(*GetResponse*) contendo os OIDs recebidos e seus valores correntes ou novos, dependendo da ação requerida (*Get/GetNext/Set*).

2.3 Evolução do Protocolo SNMP

O SNMP foi adotado como o padrão de apoio ao gerenciamento de redes interligadas pelo protocolo TCP/IP a partir de 1989. Porém, dadas as suas limitações em apoiar funções mais complexas de gerência de redes, em 1991, foi publicado um suplemento às suas especificações chamado *Remote Network Monitoring* - RMON. Esse suplemento estende o SNMP, introduzindo-lhe a capacidade de apoio ao gerenciamento de redes locais. Originalmente, o protocolo SNMP provia apenas informações sobre os nodos das redes individualmente [Stalling1999].

Ainda conforme Stallings(1999), em 1995, após ter passado por algumas melhorias funcionais, uma nova versão do padrão SNMP, até então identificada como SNMPv1, foi lançada, chamada SNMPv2. Do mesmo modo, nesse mesmo ano, 1995, as especificações RMON foram estendidas e uma nova versão, a RMON2, foi editada.

Finalmente, em 1998, o protocolo SNMPv3 foi publicado em um conjunto de documentos que definiam os recursos de segurança e uma arquitetura com suporte para futuras extensões dessa nova versão.

2.3.1 O Protocolo SNMPv2

O protocolo SNMPv2 ampliou o protocolo SNMPv1 em vários aspectos. Os principais se deram na estrutura das informações de gerenciamento, nas operações do protocolo e na introdução da capacidade de comunicação Gerente-Gerente.

A estrutura das informações de gerenciamento foi alterada de modo a comportar novos tipos de dados e melhorar a documentação dos objetos. Dentre as alterações mais significativas, encontram-se a criação de contadores de 64 bits e a adoção de uma nova convenção para a criação e exclusão de linhas em uma tabela de objetos.

No protocolo SNMPv1, em linhas gerais, a simples atribuição de valores aos índices, juntamente com a atribuição de valores aos demais objetos de uma tabela bastam para realizar a criação de uma nova linha nessa tabela. Por outro lado, a remoção de uma linha é causada por meio da atribuição de um valor específico a um objeto da tabela especialmente projetado para esse fim. As mensagens, a seguir, ilustram a manipulação de uma linha da tabela *ipRouteTable* do grupo *ip* da MIB-II.

Mensagem 1	set (ipRouteDest.10.10.10.1 = 10.10.10.1, ipRouteMetric.10.10.10.1 = 9, . . .)
Efeito 1	Causa a criação de uma linha na tabela *ipRouteTable*.
Mensagem 2	set (ipRouteType.10.10.10.1 = 2 (invalid))
Efeito 2	Causa a remoção da uma linha cujo índice é 10.10.10.1 da tabela *ipRouteTable*.

Já, no protocolo SNMPv2, também em linhas gerais, a criação e exclusão de linhas são apoiadas por um objeto da tabela cuja definição inclui as cláusulas

SYNTAX e *MAX-ACCESS* com valores *RowStatus* e *read-create*, respectivamente. Esse objeto é denominado de *status column* (*sc*) das linhas da tabela.

Nessa versão, SNMPv2, ao objeto *sc* deve-se atribuir o valor *createAndGo*(4) ou *createAndWait*(5), seguindo-se a atribuição dos valores dos índices e dos demais objetos da nova linha. Caso o valor do *sc* seja *createAndGo*, a linha é imediatamente criada e disponibilizada para o agente SNMP. Nesse instante, o valor do *sc* passa a ser *active*(1). Se *createAndWait* for o valor do objeto *sc*, uma linha é criada, mas não fica disponibilizada para o agente, pois o valor de *sc* muda para *notInService*(2), sendo então necessário que a aplicação Gerente, explicitamente, altere o valor de *sc* para *active*. A ocorrência de algum erro durante esse procedimento faz com que o valor do *sc* fique sendo *notReady*(3), indicando que a linha não está disponível para o agente. Por fim, a exclusão é realizada quando *notInService*(2) ou *destroy*(6) é atribuído ao objeto sc. No primeiro caso, a exclusão é apenas lógica, e no segundo, a linha é removida, de fato, da estrutura da tabela. Vale salientar que nem toda tabela aceita manipulação por parte de aplicações de gerenciamento. As mensagens, a seguir, ilustram essa nova forma de manipulação de linhas sobre a tabela *ifStackTable* do grupo *ifMIBObjetcs* da MIB-II do SNMPv2. Nessa tabela, o objeto *ifStackStatus* realiza a função de *status column* (*sc*).

Mensagem 1	set (ifStackHigherLayer.1.2 = 1, ifStackLowerLayer.1.2 = 2, ifStackStatus.1.2 = createAndWait)
Efeito 1	Cria um linha na tabela com valores <1, 2, 2(notInService)>. Procedimento o.k.
Mensagem 2	set (ifStackStatus.1.2 = active)
Efeito 2	Torna a linha criada, disponível para o agente SNMP

No campo das operações do protocolo SNMPv2, a alteração mais marcante foi a introdução de dois novos tipos de mensagens: a do tipo *GetBulk*, para requisitar eficientemente grandes volumes de dados, e a do tipo *Inform*, para comunicação de *Traps* entre Gerentes.

Uma nova base de informações gerenciais, a MIB SNMPv2, foi definida. Ela contém informações básicas sobre o tráfego decorrente das operações do protocolo SNMPv2 propriamente dito. Essa MIB mantém os grupos de informações originais da MIB-II, mas expande os grupos *interfaces* e *system*, remodela o grupo *snmp* e ganha um novo grupo, o grupo *MIB Objects.*

O grupo *interfaces* da MIB-II foi reestruturado com o acréscimo das tabelas *extension table*, *stack table*, *test table* e *receive address table*.

O grupo *system* original da MIB-II foi expandido com uma coleção de objetos que permite a descrição e a configuração dinâmica dos objetos do nodo instrumentalizado com um Agente SNMPv2.

A MIB SNMPv2 reformulou o grupo *snmp* original da MIB-II e introduziu outras informações relativas a configuração de Gerentes e Agentes SNMPv2. O novo grupo *snmp* consiste de objetos que permitem o acompanhamento das atividades do SNMPv2 propriamente dito.

O grupo *MIB Objects* é uma coleção de objetos que lidam com as *traps* do protocolo SNMPv2 e que permitem a coordenação das operações do tipo *Set* entre Agentes que atuam como Gerentes.

2.3.2 O Protocolo SNMPv3

O protocolo SNMPv3 provê uma série de medidas de segurança através do empacotamento das mensagens com informações de controle de acesso aos objetos gerenciados e contra o acesso ilícito às mensagens que trafegam pela rede. Dessa maneira, as mensagens do protocolo SNMPv3 passam a ter um formato completamente distinto das mensagens das versões anteriores. Ainda assim, estas estão embutidas, de modo protegido, nas mensagens desse

protocolo. A proteção das mensagens é feita através da aplicação de processos de criptografia e de autenticação sobre as mesmas.

O controle de acesso cuida para que os objetos gerenciados sejam agrupados, que esses grupos sejam acessados apenas por determinados usuários ou grupos de usuários e que as requisições desses usuários ou grupos de usuários originem-se em um determinado equipamento da rede a partir de uma determinada porta.

Para esse protocolo, também, foi padronizada uma MIB específica, a MIB SNMPv3. Ela é composta pelos grupos *snmpTargetObjects*, *snmpNotifyObjects* e *snmpProxyObjects*. A utilização em conjunto desses grupos permite a configuração das características de controle de acesso e de segurança a serem aplicadas ao processo de comunicação entre os equipamentos da rede instrumentalizados com esse protocolo.

2.3.3 Monitoração de Redes Remotas

Segundo a arquitetura do protocolo SNMP, um Gerente obtém informações apenas dos nodos individuais da rede. Para superar essa limitação, estabeleceu-se a padronização da MIB RMON. Essa nova MIB especifica uma base de informações gerenciais que parametrizam as atividades de Monitoração das Redes Remotas ou de subredes.

O processo de monitoração é executado por uma sonda que instrumentaliza a MIB RMON. Na atividade de monitoração, basicamente, a sonda permanece "escutando" as interfaces do seu nodo, conectadas às sub-redes, com o propósito de analisar os pacotes que por elas transitam. Além das análises, ações também são especificadas na MIB RMON.

A especificação da MIB RMON é composta por grupos de objetos que permitem a configuração das características da atividade de monitoração a serem aplicadas nas análises e ações executadas pela sonda.

Essa primeira versão da MIB RMON se restringia a produzir informações baseadas única e exclusivamente nos endereços *Medium Access Control* - MAC contidos nos *frames* que transitam nas subredes monitoradas através de interfaces do tipo Ethernet e TokenRing.

Para superar essa restrição, novos grupos de objetos foram acrescentados a MIB RMON, surgindo, assim, a segunda versão dessa MIB ou MIB RMONv2 cuja implementação é feita por uma sonda.

Uma sonda RMONv2 pode monitorar e decodificar informações relativas aos protocolos situados acima da camada MAC, isto é, ela é capaz de "ver" informações empacotadas pelos Protocolos IP, TCP, UDP, SNMP e outros mais situados nos vários níveis acima da camada MAC. Isso significa que essa sonda pode produzir informações, agora, baseadas em combinações de identificações de protocolos, endereços IP e portas, todas relativas aos *hosts* das subredes monitoradas.

Em geral, os objetos das MIBs RMON e RMONv2 são mantidos em dois tipos de tabelas: uma tabela de controle e uma ou mais tabelas de dados. A tabela de controle, normalmente, descreve as tabelas de dados, isto é, especifica quais, quantas e como as informações estarão contidas nas tabelas de dados. Uma aplicação Gerente define os parâmetros do processo de monitoração, alimentando as tabelas de controle das sondas RMON ou RMONv2.

2.4 Tendências para o Futuro

O gerenciamento de redes de computadores baseado no protocolo SNMP encontra-se em constante estado de evolução através dos trabalhos

desenvolvidos pela Força Tarefa de Engenharia da Internet (IETF - Internet Engineering Task Force).

De acordo com [Case 2001], atualmente a IETF está desenvolvendo esforços nas seguintes áreas:

- Gerenciamento de Configuração baseado no protocolo SNMP;
- Protocolo SNMP propriamente dito;
- Estrutura das informações gerenciais;
- Gerenciamento Distribuído;
- Novas MIBs.

Na área de Gerenciamento de Configuração, a tônica gira em torno do suporte à garantia da configuração de políticas de segurança baseadas em regras. Para tanto, faz-se necessária a especificação de uma MIB que se alimente de várias instâncias de parâmetros altamente abstratos, que sejam independentes de fabricantes e tecnologias, e que integre a configuração com a gerência de falhas, desempenho, monitoração, etc.

Essa MIB deve alavancar as ferramentas e infra-estrutura existentes, assimilar políticas do mundo real, suportar regras de políticas, etc. Tudo isso por meio da codificação de *scripts* em linguagem bastante simplificada.

Na área do protocolo SNMP propriamente dito, o IETF está desenvolvendo esforços para produzir a próxima geração do *framework* para sistemas de gerenciamento baseado em SNMP. Nessa área, eles estão buscando suprimir e comprimir eficientemente os OIDs, melhorar a manipulação de tabelas e de operação de filas e suportar novos tipos de dados.

No tocante à estrutura das informações gerenciais, o IETF está desenvolvendo uma proposta para a nova linguagem de definição de dados, totalmente independente de protocolos, que suporte inteiros de 64 bits positivos e

negativos, ponto flutuante, uniões, vetores, agregação de tipos de dados, orientação a objetos e restrições. Para essa linguagem, buscam uma gramática cuja sintaxe seja similar à sintaxe da linguagem C.

Na área de gerenciamento distribuído, com as garantias proporcionadas pela configuração de políticas de segurança, o IETF busca a possibilidade de ter agentes inteligentes fazendo gerenciamento distribuído. Para tanto, estão desenvolvendo a padronização da Distribuição e a especificação de MIBs para Aplicações de Gerenciamento Distribuído. Com a distribuição dessas aplicações, a carga de tráfego na rede com mensagens SNMP diminui, pois o *polling* passa a ser local, isto é, a aplicação que coleta dados é executada no próprio equipamento da rede gerenciado. Nessa área, foram publicadas as MIBs de Escalonamento (RFC2591 - execução dirigida por temporização), de *Scripts* (RFC2592 - movimentação de *scripts*), de Operação Remota (RFC2925 - *ping, traceroute e DNS lookup*), de Eventos (RFC3014 - ações baseadas em *thresholds*), e de Registro de Notificações (RFC3014 - *log notofication*).

Na área das MIBs, novas especificações para outros padrões foram criadas. Por exemplo: a MIB WWW, a MIB de Aplicação, a MIB de Sistemas de Aplicações, a MIB de Monitoração de Serviços da Rede e a MIB de Recursos do *Host*. Elas permitem que um administrador gerencie os serviços e os servidores tanto seus quanto dos seus clientes.

2.5 Conclusão

O cerne do *Simple Network Management Protocol* é o protocolo propriamente dito, estabelecido de forma padronizada, para troca de mensagens entre uma aplicação Gerente e os agentes SNMP. Além disso, dentro do contexto da heterogeneidade dos nodos, outro aspecto central desse protocolo é a abstração das especificidades de *hardware* e *software* que é provida por meio da sua MIB às aplicações Gerentes.

Embora o nome desse protocolo sugira que ele exerce alguma função de gerenciamento, isso não é de todo uma verdade, pois quem realiza ações gerenciais são os Sistemas de Gerenciamento de Redes de Computadores.

Capítulo 3 - Sistemas de Gerenciamento de Redes de Computadores

3.1 Introdução

Conforme a Organização Internacional para Padronização (*International Organization for Standardization* - ISO), um sistema de gerenciamento de redes deve administrar a complexidade, a qualidade do serviço, o balanceamento das necessidades, o tempo de manutenção e o custo das redes. Para tanto, ela definiu as funcionalidades juntamente com os requisitos (como alcançar a funcionalidade) que um sistema de gerenciamento deve prover. A ISO distribuiu essas funcionalidades em 5(cinco) áreas sobre as quais a gerência deve atuar, são elas [Stallings1999]:

1. Gerência de Falhas;
2. Gerência de Contabilidade;
3. Gerência de Nomes e Configuração;
4. Gerência de Desempenho;
5. Gerência de Segurança.

A Gerência de Falhas deve facilitar a detecção, o isolamento e a correção de anomalias na rede. Requer a realização do rastreamento e do controle de falha de modo rápido e eficaz.

A Gerência de Contabilidade deve facilitar a contabilização do uso dos recursos da rede, permitindo o rastreio contábil por usuário ou grupos de usuários dessa utilização. Requer o rastreio do uso dos recursos buscando: abuso de utilização dos recursos, uso ineficiente dos recursos e conhecimentos detalhados das atividades dos usuários.

A Gerência de Nomes e de Configuração deve facilitar a identificação e o controle dos objetos gerenciados. Requer a coleta e a atribuição de valores aos

objetos gerenciados, visando o funcionamento contínuo das interconexões dos serviços.

A Gerência de Desempenho deve facilitar a avaliação do comportamento dos objetos gerenciados e da eficácia das atividades de comunicação. Requer a associação de métricas aos objetos gerenciados. Ela compreende duas categorias funcionais bastante abrangentes: Monitoração e Controle. A Monitoração rastreia atividades na rede e o Controle permite que ajustes sejam feitos para melhorar o seu desempenho.

A Gerência de Segurança deve tratar da proteção das informações dos objetos gerenciados e da proteção do acesso às atividades de controle da gerência. Requer a geração, distribuição e armazenamento de chaves e algoritmos de criptografias. Autenticações, senhas e outras informações de controle de acessos também devem ser mantidas e distribuídas.

3.2 Gerenciamento Baseado em Eventos

Após uma análise de vários trabalhos na área de gerenciamento de redes, é possível verificar que as abordagem utilizadas, em geral, buscam identificar e perceber Eventos, como meio para detecção, diagnóstico e correção de anomalias das redes de computadores. Essas abordagens empregam, por exemplo, cálculos probabilísticos, grafos e raciocínios baseados em regras e em casos.

As abordagens probabilísticas são empregadas no controle da Qualidade de Serviço - QoS e das reservas de recursos em redes ATM [Li 2000], na produção de alarmes de previsão de falhas [Thottan 1999], na identificação de "assinaturas" do comportamento dos objetos gerenciados a partir do tráfego na rede [Papavassiliou 1998], e na antecipação de problemas potenciais em um servidor Web [Shen 2000].

Grafos de causalidade e de dependência são empregados na determinação das causas básicas, de uma cadeia de alarmes ou eventos [Lo 1998] [Yemini 1996] e [Shwartz 2000].

Regras são formuladas como *thresholds* cujas definições baseiam-se em séries temporais de estados dos objetos da rede [Ho 2000].

Raciocínio baseado em casos é aplicado em um sistema de apoio à manutenção de uma rede de computadores, auxiliada por uma base de Conhecimento de Eventos [Burges 2000] e na integração de redes heterogêneas e correção das falhas dos seus nodos [Penido 1999]. Essa técnica também foi integrada à arquitetura *Trouble Tickets Systems* - TTS para propor soluções de falhas, baseadas em episódios passados ocorridos em uma rede de computadores [Melchiors 2000].

Em síntese, as aplicações desenvolvidas nesses trabalhos são alimentadas por alarmes ou arquivos de *logs*. Elas buscam, a partir dos eventos, antever tendências e anomalias, detectar, isolar e corrigir as causas originais dos eventos e mapear as observações físicas com os estados dos objetos gerenciados da rede. A utilização de *thresholds*, métodos estatísticos e grafos apresentam como contrapartida, por exemplo, mais algoritmos nos roteadores, modelos matemáticos e estatísticos complexos, mais camadas nos modelos de gerenciamento, etc.

Atualmente, a quantidade e os tipos de aplicações de gerenciamento de redes são grandes e suas características computacionais variam bastante. Infelizmente, o desenvolvimento dessas aplicações não leva em conta as necessidades e os domínios de gerenciamento em geral [Hariri 2000]. Isso induz a necessidade de um mecanismo, genérico e automático, de modelagem capaz de representar tais domínios de modo a facilitar a detecção de eventos. Nesse caso, Domínio é uma situação específica do mundo real das redes de computadores, expressa por meio de um agrupamento lógico de objetos SNMP.

3.3 Monitoração no Protocolo SNMP

Diante dos requisitos para consecução do gerenciamento de rede de computadores preconizado pela ISO, é possível perceber que a monitoração é a principal atividade de um sistema de gerenciamento.

As aplicações de gerenciamento de redes consistem, principalmente, na monitoração, interpretação e manipulação de eventos. Em geral, os eventos são definidos como condições anômalas observadas no funcionamento das redes. Eles, normalmente, são problemas que ocorrem no hardware e/ou software dos nodos da rede [Yemini 1996]. No cenário do protocolo SNMP, esses eventos ou condições podem ser detectados por meio das variações ocorridas nas informações gerenciais das MIBs dos Agentes e/ou das sondas RMON.

Essas informações gerenciais, quando providas por um Agente, são obtidas como visões instantâneas no tempo, e no caso das sondas RMON, são obtidas como grupos de informações consolidadas ao longo de um período de tempo. Em ambos os casos, a aplicação Gerente fica encarregada de requisitar, periodicamente, as informações que necessita para analisar e executar alguma ação de controle sobre os nodos gerenciados.

A consolidação das informações das sondas RMON e RMON2 é realizada por meio da seleção de pacotes juntamente com a filtragem dos seus conteúdos. Isso objetiva gerar estatísticas e/ou eventos, em conformidade com as definições de eventos das próprias sondas. Esses eventos podem causar o disparo de *TRAPS* que enviam informações da MIB RMON/RMON2 para algum Gerente(grupos *alarm*, *filter*, *capture* e *event* das MIBs RMON e RMON2).

3.3.1 As Fontes de Dados das Aplicações de Gerência

No âmbito da gerência apoiada pelo protocolo SNMP, os sistemas são alimentados por informações oriundas das MIBs dos Agentes e/ou das sondas RMON.

Cada MIB é um conjunto geral de informações que mudam com o passar do tempo. É necessário que essas informações sejam agrupadas, a partir de supostos relacionamentos, para que um Domínio de Gerenciamento possa ser obtido. O relacionamento mais imediato que se pode perceber entre as informações é o tempo.

As sondas RMON e RMON2, de acordo com [Stallings 1999], utilizam *buffers* circulares. Conseqüentemente, a aplicação Gerente deve ter um controle mais apurado sobre os *pollings* para que o preenchimento dos *buffers* não "virem", evitando assim a perda de informações anteriormente consolidadas. As sondas, também, são computacionalmente oneradas com a execução de processos de geração de estatísticas, de eventos e traps. Esses *buffers* e processos poderiam ser deixados a cargo de outra entidade de processamento. Com isso, as sondas ficariam mais livres para, apenas, "escutar" as interfaces e filtrar pacotes, considerando-se o fato de que as redes estão ficando cada vez mais rápidas.

3.3.2 Um Modelo Genérico de Monitoração

Diante das considerações feitas sobre as fontes de dados das aplicações de gerenciamento e da diversidade dos domínios por elas gerenciados, esse trabalho, no campo da monitoração, produz uma plataforma de suporte ao desenvolvimento, apoiado pela modelagem automática de dados e de eventos, de programas dirigidos para o gerenciamento de domínios que podem surgir no cenário das redes de computadores monitoradas por meio do protocolo SNMP.

Essa plataforma permite agrupar informações oriundas dos diversos nodos das redes instrumentalizados com Agentes SNMP. Esse agrupamento resulta na criação, de modo transparente, de um repositório para essas informações em um banco de dados. Em outras palavras, ela permite a modelar estruturas de dados que representam domínios de gerenciamento, com vistas a alimentar processos de gerenciamento de redes.

Atributos datadores, também, são automaticamente incluídos nesses modelos de dados, indicando os momentos de requisição e de recepção das informações dos domínios modelados. Eles provêem informações que podem auxiliar procedimentos de análises estatísticas, de desempenho e de comportamento histórico da rede, por exemplo.

Regras, que balizam as possíveis variações das informações agrupadas, podem ser definidas e, eventualmente, utilizadas na formulação de expressões que podem representar vários tipos de eventos. Adicionalmente, visões, baseadas nessas expressões, podem ser definidas, dinamicamente, sobre esses repositórios, provendo um mecanismo de detecção desses eventos. Os possíveis processos de gerenciamento podem consultar essas visões para detectar, diagnosticar e corrigir alguma anomalia no domínio de gerenciamento monitorado.

A plataforma construída disponibiliza um agente monitorador que, automaticamente, reconhece, monitora e armazena as informações dos domínios de gerenciamento. Desse modo, as aplicações podem ficar dedicadas exclusivamente aos procedimentos gerenciais, isentando-se dos *pollings*, e apenas tomando decisões a partir da detecção dos eventos implementados como visões do banco de dados ou a partir de outras análises feitas sobre as informações dos repositórios. Assim, inúmeras pequenas aplicações podem, cada uma, gerenciar seu próprio domínio, facilitando o desenvolvimento de um Sistema de Gerenciamento mais abrangente. A plataforma tem como base as seguintes tecnologias:

- Modelagem de Domínios;
- Modelagem de Eventos;
- Modelagem de Banco de Dados;
- Modelagem Web e
- Modelagem Cliente/Servidor.

3.3.2.1 Modelagem de Domínios

Todos os sistemas de gerenciamento são modelados com base no modo como se quer que eles funcionem e pelos próprios componentes do modelo. Cada situação específica do mundo real, que é tratada por esses sistemas, é chamada de Domínio do Gerenciamento. Essa noção abstrata pode ser usada para delimitar as fronteiras físicas e administrativas das ações de gerenciamento em uma rede de computadores. Num sistema de gerenciamento modelado a partir do padrão SNMP, seus componentes são: os nodos gerenciados, pelo menos uma estação de gerenciamento, os objetos dos nodos e o protocolo SNMP propriamente dito. Esses componentes podem ser combinados e configurados de várias maneiras e de formas bastante complexas [Perkins 1997], formando diversos Domínios de Gerenciamento. A Figura 3.1 representa graficamente a idéia de formação de domínios de acordo com o conceito de Perkins.

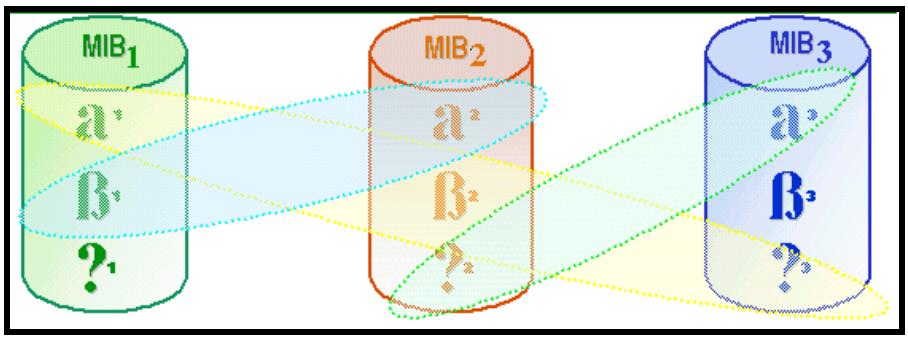

Figura 3.1 - Representação gráfica de Domínio de Gerenciamento

Os cilindros e os símbolos $\mathbf{a}_n$, β_n e $\mathbf{?}_n$ da Figura 3.1 representam, respectivamente, os nodos gerenciados com a suas respectivas MIBs (MIB_1, MIB_2 e MIB_3) e os objetos componentes dessas MIBs. Cada combinação desses componentes, como indicado pelas elipses, determina um Domínio de Gerenciamento.

3.3.2.2 Modelagem de Eventos

Eventos representam alterações, gerencialmente de interesse, que podem ocorrer em um sistema. Por exemplo, o estado de uma interface, sendo *on-line* ou *off-line*, pode ser definido como um Evento de interesse dentro de um sistema de gerenciamento de redes. Eventos primitivos são predefinidos em um ambiente e o mecanismo para sua detecção está embutido na implementação do sistema. Eventos compostos são formados pela conjugação de Eventos primitivos ou de outros Eventos compostos. Normalmente, um modelo específico é dedicado à detecção das ocorrências desses tipos de Eventos [LIU 1999]. A Figura 3.2 apresenta a pilha de conceitos usada para representar o Modelo de Eventos aplicada aos Domínios de Gerenciamento.

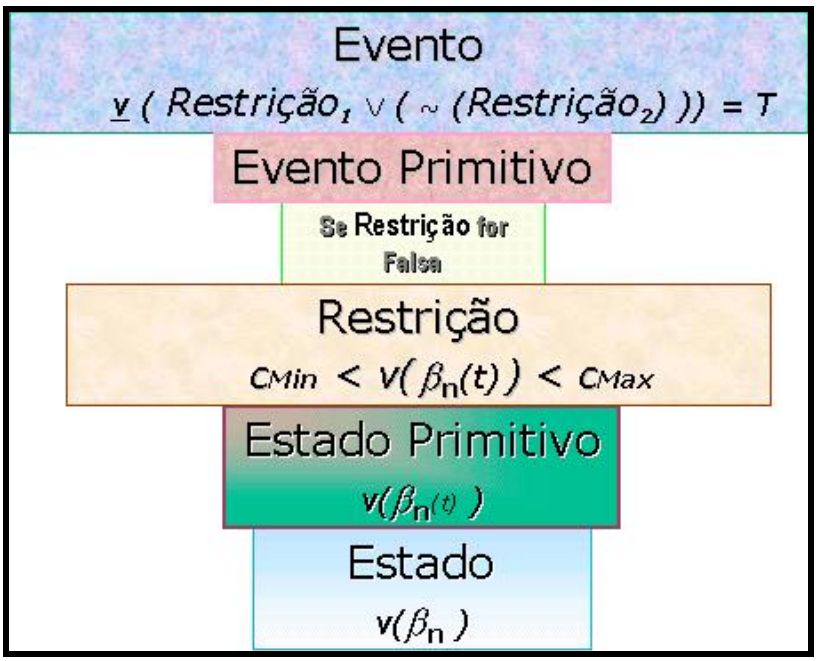

Figura 3.2 - Pilha de conceitos aplicada na Modelagem de Eventos

Nos níveis mais baixos, encontram-se os conceitos de Estado, valor de um objeto (β_n) gerencialmente de interesse, e de Estado Primitivo, valor desse objeto, tomado um instante *t*. Nos níveis mais acima, estão os conceitos de Restrição e Evento. A violação da Restrição, determinada por um intervalo de valores, indica a

ocorrência de um Evento Primitivo. Por fim, a equação elaborada com o auxílio de Restrições representa um Evento Composto ou um **Evento** propriamente dito. A avaliação lógica dessa equação, sendo verdadeira, indica a ocorrência do Evento.

3.3.2.3 Modelagem de Banco de Dados

Quase todas as aplicações desenvolvidas para apoiar procedimentos operacionais são sistemas baseados em informações. A coleta, organização, processamento e relato dos dados é fundamental para automação dessas aplicações [Swanson apud Papavasiliou 1998].

Todo sistema de gerenciamento de eventos deve ter a capacidade de modelar e armazenar as informações da rede e de seus Eventos. Isso inclui os próprios conhecimentos dos operadores da rede. Adicionalmente, esse sistema deve prover algoritmos de análise dos estados dos nodos gerenciados que detectem eventuais anomalias da rede. Por fim, é desejável que tanto a modelagem das informações quanto os processos de diagnóstico e reação sejam capazes de acompanhar o crescimento do tamanho e da complexidade das redes [Yemini 1996].

Desse modo, um Sistema Gerenciador de Banco de Dados Relacional, dotado de recursos como *Views*, *Triggers* e *Stored Procedures*, mostra-se como um candidato natural à ferramenta de apoio ao desenvolvimento de aplicações de gerenciamento de redes com as características citadas por Swanson e Yemini. Consoantemente, pode-se perceber que as idéias ilustradas pelas Figuras 3.1 e 3.2 podem desembocar em definições de estruturas de um banco de dados relacional.

3.3.2.4 Modelagem *Web*

Desde há algum tempo, novas tecnologias surgiram para a *Web*. Atualmente, além da linguagem Java, estão disponíveis as tecnologias Java *servlets* e *Remote*

Method Invocation - RMI, que permitem o projeto de novos modelos de aplicações de gerenciamento de redes [Martin 1999].

As aplicações de gerenciamento baseadas em interfaces *Web* são mais fáceis de usar do que as que possuem interfaces baseadas em linha de comando. Elas podem ser usadas a partir de qualquer computador ou estação de trabalho com um *browser*. Isso significa que profissionais de planejamento, projetistas e gerentes não precisam usar aplicações clientes especializadas nos seus *laptops*. Com um *browser*, eles podem acessar a rede a qualquer instante e de qualquer lugar onde estejam [Muller 1997].

As Interfaces *Web* permitem que os desenvolvedores criem aplicações Cliente/Servidor, simples e poderosas, virtualmente acessíveis de quaisquer plataformas [Deri 1999], com o auxílio de um servidor *Web* que suporte Java *servlets*.

3.3.2.5 Modelagem Cliente/Servidor

Sistemas Cliente/Servidor são aplicações cuja arquitetura compreende, basicamente, duas partes lógicas: um servidor que provê serviços e um cliente que requer os préstimos de um servidor. Juntas formam um sistema completo na qual existe uma clara divisão de responsabilidades. Esse modelo é chamado de Cliente/Servidor de Duas Camadas. Também é possível que uma aplicação tenha as interfaces do usuário, a lógica e a sua base de dados, todas separadas. Nesse caso, tem-se um modelo Cliente/Servidor de Três Camadas [Lewandowski 1998].

3.4 Conclusão

O monitoramento, isto é, a coleta sistemática de dados é a principal atividade de um sistema de gerenciamento de redes de computadores. O objetivo básico da monitoração é obter informações sobre o comportamento e sobre os estados dos nodos gerenciados.

Atualmente, uma das abordagens utilizadas no desenvolvimento de sistemas de gerenciamento é dotá-las com a capacidade de percepção de eventos. Para isso, esses sistemas utilizam técnicas como análises estatísticas, teoria dos grafos, raciocínio baseado em regras ou em casos, etc.

No campo do Monitoramento e da percepção de Eventos, não é difícil perceber que a articulação das tecnologias de Bancos de Dados Relacionais, Cliente/Servidor, Web e Modelagem de Domínios e Eventos pode produzir um *framework* que facilite o desenvolvimento de aplicações de gerenciamento de redes, capaz de prestar serviços automáticos de coleta e disponibilização de informações gerenciais, provendo também mecanismos de detecção on-line de Eventos.

Capítulo 4 - SGME, Um Sistema Genérico de Monitoração de Redes de Computadores Dirigido a Eventos

4.1 Introdução

Esse capítulo apresenta as especificações de um Sistema Genérico de Monitoração de Redes de Computadores Dirigido a Eventos - SGME. Ele é um *framework* de apoio a construção de sistemas de gerenciamento de redes de computadores, provendo:

- interface operacional *Web*;
- facilidades para definição de domínios de gerenciamento;
- abstração do processo de coleta e armazenamento de dados;
- abstração do protocolo SNMP nas fases não reativas do processo de gerenciamento(comunicação, estrutura de dados, etc.);
- facilidades para automatização da coleta, datação e armazenamento de informações;
- facilidades para automatização da percepção de Eventos;

A partir das especificações apresentadas, será montado um *framework* aberto, modelado com componentes padronizados. Ele é construído por meio da integração do protocolo SNMP com Bancos de Dados Relacionais e provê as seguintes funcionalidades:

- comunicação entre o administrador e os nodos gerenciados da rede;
- definição de domínios baseados nos estados dos nodos da rede;
- monitoração dos nodos da rede;
- construção dinâmica e transparente de repositórios para os dados monitorados;
- definição de eventos;
- produção dinâmica e transparente de relatórios gerenciais.

4.2 A Arquitetura do SGME

O Sistema Genérico de Monitoração de Redes de Computadores Dirigido a Eventos - SGME - é um *framework* formado por componentes tecnológicos desenvolvidos nas áreas de Sistemas Operacionais, Sistemas Web, Sistemas de Bancos de Dados, Interfaces Gráficas, Linguagens de Programação e Protocolos de Comunicação e de Gerenciamento de Redes de Computadores. A abstração de *hardware* e de sistemas operacionais são as características que motivam a escolha desses componentes. A existência de componentes com essas características permite a construção do SGME em conformidade com a Arquitetura Padrão Cliente/Servidor de n Camadas. No caso do SGME, três camadas.

A arquitetura do SGME é composta por um ou mais Clientes formando a primeira camada, um Servidor de Monitoração na segunda camada e um ou vários Servidores de Informações Gerenciais (Agentes SNMP) na terceira camada, conforme mostra a Figura 4.1.

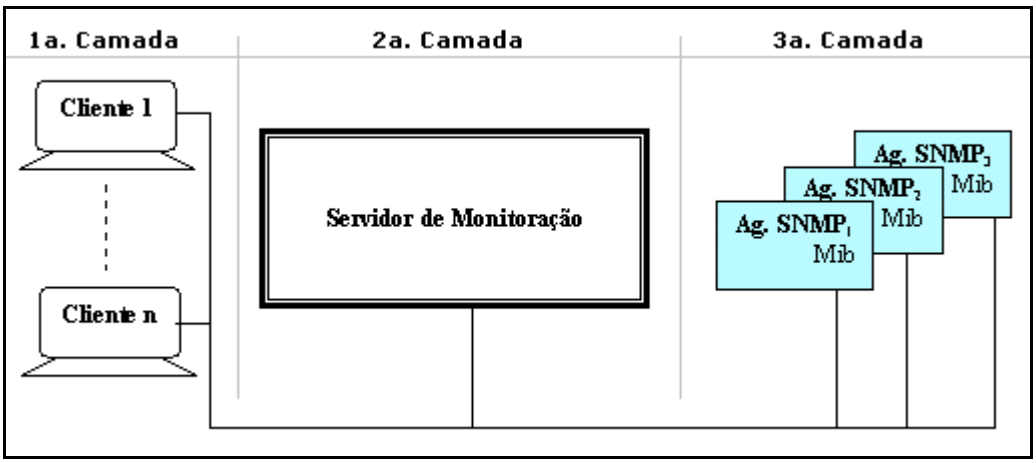

Figura 4.1 - Arquitetura Cliente/Servidor do SGME

4.2.1 A Primeira Camada - Camada dos Clientes

Um Cliente é um computador com algum navegador de Internet com suporte à Máquina Virtual Java (Java Virtual Machine - JVM). Um *Browser* com suporte a

JVM é bastante para prover a comunicação entre a primeira e segunda camadas da arquitetura, oferecendo, adicionalmente, uma interface gráfica moderna que possibilita a interação entre o Administrador da rede e ambos, o Servidor de Monitoração e o Banco de Dados.

O objetivo dessa camada é permitir que o administrador proceda, dinamicamente, a configuração do processo de monitoração dirigida a eventos, além de definir, produzir, consultar e analisar relatórios sobre estados e eventos históricos, armazenados em um banco de dados relacional.

4.2.2 A Segunda Camada - O Servidor de Monitoração

O Servidor de Monitoração pode ser qualquer computador com um Servidor Web que suporte Java Servlets e JDBC (Java Database Connectivity). Precisa, também, suportar qualquer Sistema Gerenciador de Banco de Dados Relacional(SGBDR). Para a implementação do protótipo do SGME, foram adotados o Servidor Apache Tomcat versão 4.0.3 da ASF(Apache Software Foundation), o SGBDR MyQSL da MySQL AB e Java servlets com APIs JDBC e AdventNet.

O servidor Apache foi selecionado por possuir *containers* para classes servlets, além de representar uma das plataformas mais utilizadas nas áreas de Sistemas Web [NETCRAF 2002], podendo ser encontrado gratuitamente no *site Apache Jacarta Project* [TOMCAT 2002], existindo em versões para as plataformas como Microsoft, Sun e Macintoch. O SGDBR MySQL foi selecionado devido ser o servidor de banco de dados que está disponível gratuitamente na Internet [MYSQL 2002] sob a licença para uso público geral em trabalhos sem fins lucrativos. Finalmente, selecionamos os Java Servlets com APIs JDBC de manipulação de bancos de dados e APIs AdventNet de comunicação com o protocolo SNMP tanto por representarem tendências nas áreas das Linguagens de Programação, da Distribuição de Serviços e da Conectividade com Bancos de

Dados como pela gratuidade com que são encontrados na Internet[ADVENTNET 2002].

O objetivo dessa camada é suportar os serviços do SGME. Esses serviços, implementados com o auxílio de GUIs(Graphical User Interface), provêem, tanto a comunicação entre o Administrador e o *framework*, visualmente manifestadas como páginas Web codificadas como formulários HTML, como as funcionalidades básicas do SGME. As informações desses formulários são trocadas com o Servidor Web por meio do protocolo HTTP(Hiper Text Transfer Protocol) onde são processados por classes Java servlets. A simples referência à URL(Universal Resource Locator) http://<servidor_de_monitoração>[:8080]/SGME do Servidor Web feita por um cliente estabelece o início da interação com o SGME.

Além dos componentes Apache Tomcat, MySQL, APIs JDBC e AdventNet, o *framework* também possui um Agente Monitorador que executa o processo de monitoração dos Domínios. As bases funcionais desse processo estão permeadas pela definição do Modelo de Dados Genérico Dirigido a Eventos (MDGE) para monitoração de redes de computadores.

4.2.3 A Terceira Camada - A Camada dos Servidores de Informações Gerenciais

A terceira camada da arquitetura do SGME é composta pelos nodos da rede, instrumentalizados com Agentes SNMP e o seu objetivo é prover informações gerenciais para o SGME. Nesse caso, os Agentes fazem o papel de Servidores de Informações de Gerenciamento, provendo os valores dos Objetos Gerenciados que se encontram nas suas MIBs. Isso significa que o Servidor da segunda camada faz o papel de Cliente de Informações de Gerenciamento. A Figura 4.2 mostra a arquitetura geral do SGME.

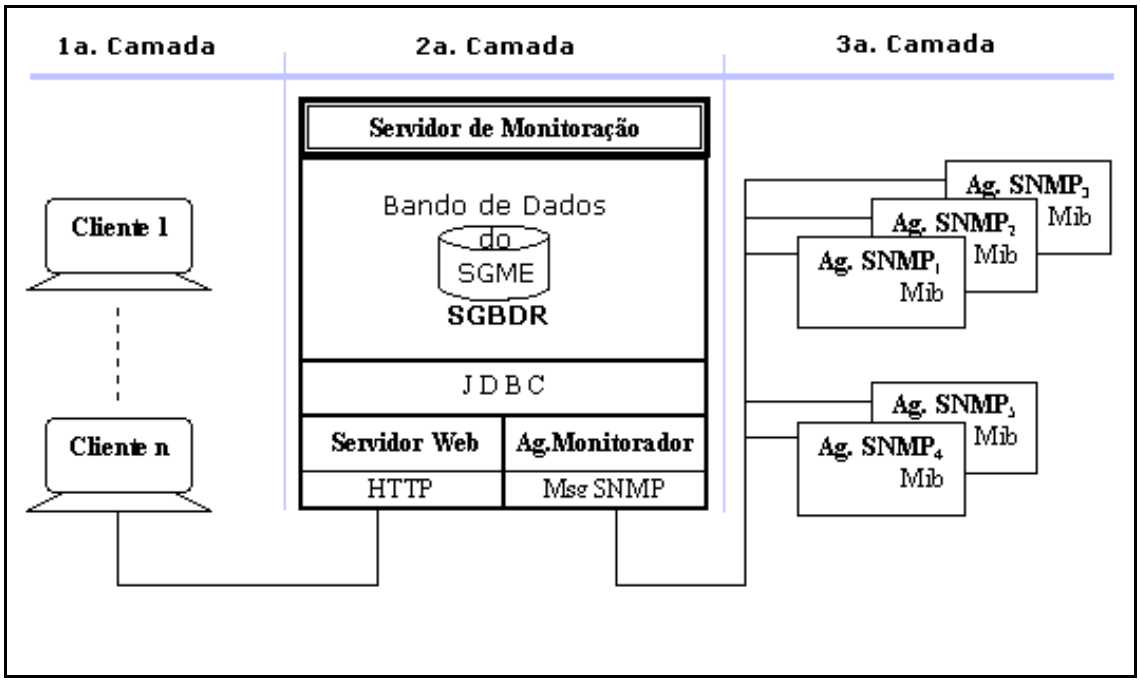

Figura 4.2 - Arquitetura Geral do SGME

4.2.4 O Banco de Dados do SGME

O Banco de Dados do SGME foi definido a partir do Modelo de Dados Genérico Dirigido a Eventos (MDGE). Ele é constituído pelas estruturas de apoio ao processo de monitoração, pelas informações obtidas pelo Agente Monitorador e pelos mecanismos utilizados para a detecção de Eventos. Nesse banco, dados e eventos históricos, oriundos dos Agentes SNMP, serão armazenados como séries temporais. Desse modo, o Banco de Dados poderá apoiar a produção de relatórios gerenciais sobre o comportamento da rede. Ele, também, pode apoiar outras atividades de gerenciamento da rede não providas pelo SGME.

4.2.4.1 O Modelo de Dados Genérico Dirigido a Eventos - MDGE

O Modelo de Dados Genérico Dirigido a Eventos (MDGE) representa um conjunto de entidades representadas por Tabelas cujos atributos e relações especificam quais, onde, quando e como os Objetos gerenciados são coletados e armazenados pelo Agente Monitorador. Tais especificações representam as

necessidades de monitoração do Administrador. Além disso, esse Modelo é dirigido a eventos, isto é, ele permite detectar as variações dos valores coletados a qualquer instante por meio de uma determinada regra restritiva desse valor.

Algumas entidades desse Modelo são criadas, transparente e dinamicamente, em função da configuração do processo de monitoração previamente definida pelo Administrador. Elas são os repositórios dos valores dos Objetos SNMP Gerenciados. Os registros neles armazenados, como dito anteriormente, podem ser considerados como séries temporais por meio das quais análises estatísticas históricas mais apuradas podem ser executadas.

O MDGE é composto pelas seguintes entidades: Comunidade, Domínio, EstadoPrimitivo, EstadoComposto, Índice, Fórmula e TabelaGeral. Também fazem parte do Modelo um número indeterminado de entidades chamadas de Repositórios. Cada Repositório é implementado como uma Tabela cujo nome é parcialmente herdado do Domínio que representa. A Figura 4.3 mostra uma visão geral do Modelo Entidade-Relacionamento do MDGE.

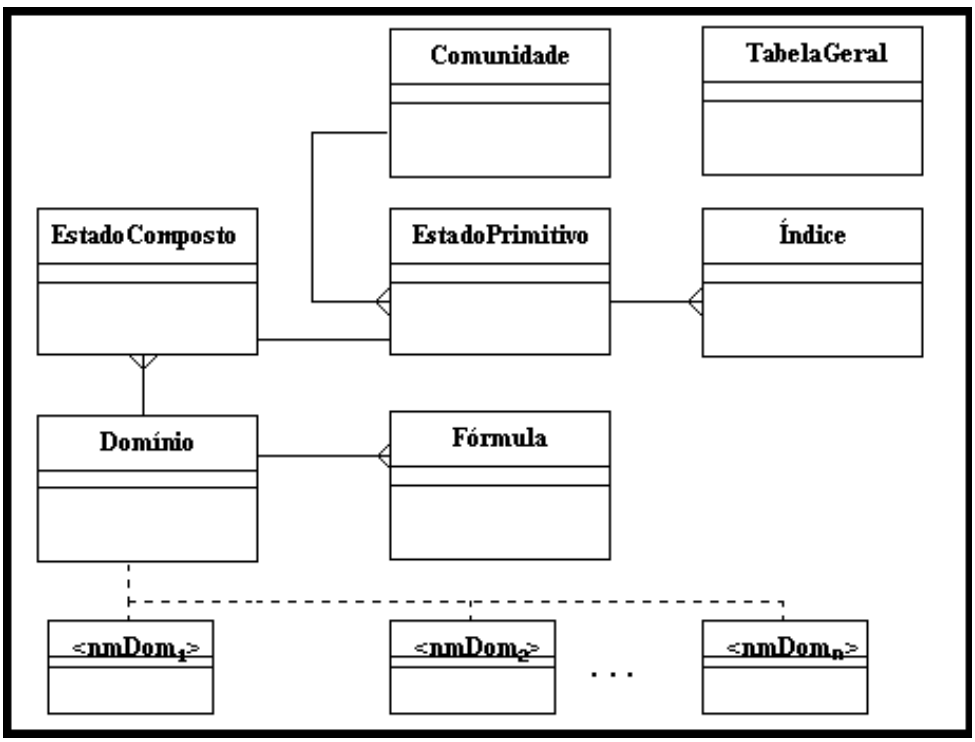

Figura 4.3 - Visão Geral do Modelo Entidade-Relacionamento do MDGE

A Entidade Comunidade representa os dados necessários para o estabelecimento de conexão entre o SGME e o Agente Monitorador, ambos com os Agentes SNMP.

As Entidades EstadoPrimitivo e Índice representam os dados necessários para se referenciar aos objetos nas MIBS dos Agentes SNMP.

As Entidades Domínio e EstadoComposto representam o agrupamento de estados primitivos em domínios de monitoração.

A Entidade Fórmula representa as restrições que são implementadas como visões ou *views* que detectam a ocorrência de eventos a partir do dados armazenados nos Repositórios.

Os Repositórios representam os Domínios de Gerenciamento cujas informações são efetivamente coletadas e armazenadas pelo Agente Monitorador. Os seus atributos são definidos a partir dos dados armazenados na tabela EstadoPrimitvo. Isso é feito com base nos relacionamentos estabelecidos entre as entidades Domínio, EstadoComposto e EstadoPrimitivo.

A Entidade TabelaGeral representa uma tabela genérica, isto é, uma tabela de tabelas, cujo objetivo é manter a simplicidade estrutural do MDGE. Nesse sentido, todas as demais informações de caráter acessório, sem relação direta com a configuração do processo de monitoração, ao SGME são modeladas como tabelas de forma tal que possam ser univocamente identificadas, armazenadas e recuperadas.

4.3 As Funcionalidades do SGME

O *framework* é composto por três grupos funcionais de serviços que permitem ao SGME alcançar os seus objetivos. O primeiro grupo configura o

processo de monitoração dos nodos da rede e define os Eventos. O segundo define e gera dinamicamente relatórios gerenciais a partir dos dados armazenados nos seus repositórios. E o último grupo, formado apenas pelo Agente Monitorador, executa o processo de monitoração propriamente dito.

Os primeiro e segundo grupos funcionais ficam a cargo do Servidor Web Apache Tomcat. O primeiro grupo funcional é representado pelos serviços de Registro de Comunidades, de Estados Primitivos, de Indexação, de Domínios, de Controle da Coleta e de Controle de Eventos. O segundo grupo funcional é responsável pela geração de relatórios por meio dos serviços de Visualização dos Repositórios e dos Eventos. Esses serviços são realizados por Java *servlets* que produzem formulários HTML como interfaces entre o SGME e o Administrador da rede. Eles interagem com o banco de dados por meio de APIs JDBC, configurando o processo de monitoração e produzindo relatórios.

O último grupo funcional fica a cargo do Agente Monitorador que pode ser executado no mesmo computador do Servidor Web Apache Tomcat, como pode ser visto na Figura 4.2. Pode, também, ser executado em qualquer computador com o ambiente Java *RunTime*, com a biblioteca de classes AdventNet e com a ferramenta de definição de Fontes de Dados ODBC com as devidas interfaces para o banco de dados em uso. Ele é representado pelos procedimentos de reconhecimento dos domínios e de monitoração e armazenamento das informações coletadas.

4.3.1 A Configuração do Processo de Monitoração

A configuração do processo de monitoração permite ao Administrador definir os Domínios que deseja monitorar. Essas definições são feitas através dos serviços: Registro de Comunidades, Registro de Estados Primitivos, Registro de Indexação, Registro de Domínios, Registro de Controle da Coleta e Registro de Eventos.

O serviço Registro de Comunidades mantém as informações que permitem ao Agente Monitorador requisitar dados aos Agentes SNMP.

O serviço Registros de Estados Primitivos especifica os Objetos sobre os quais podem recair as atenções do administrador da rede. Ele também pode estabelecer Restrições para esses Objetos. Complementarmente, no caso do registro de Objetos da forma colunar, a transação de Registro de Indexação permite a especificação dos seus índices.

O serviço Registro de Domínios agrupa os Estados Primitivos que, segundo a visão do Administrador, estão logicamente relacionados. Cada grupo é chamado de Domínio de monitoração, e os valores dos seus componentes são armazenados em repositórios. Um para cada Domínio.

O serviço Registro de Controle da Coleta dispara um processo de monitoração para cada um dos Domínios definidos, ao mesmo tempo em que procede a criação dos seus repositórios.

O serviço Registro de Eventos define visões do banco de dados ou *views* sobre os Domínios previamente registrados. Desse modo, a detecção de eventos fica a cargo do Sistema Gerenciador de Banco de Dados do SGME.

4.3.2 A Definição e a Geração de Relatórios Gerenciais

O MDGE dá condições para a construção dos repositórios dos dados obtidos pelos processos de monitoração. Com isso, os serviços de definição e geração de relatórios podem gerar formulários HTML, previamente preenchidos com uma lista de atributos desses repositórios que podem ser selecionados, de modo muito simples, para compor relatórios. Filtros temporais para esses relatórios também podem ser especificados pelo Administrador. Isso é feito de forma bastante amigável.

Além das definições das estruturas dos repositórios, acham-se disponíveis, também no MDGE, as restrições, Visões, que se aplicam aos objetos monitorados. Assim, as transações de Definição e Geração de Relatórios Gerenciais podem consultar as visões para relatar os eventos ocorridos na rede em um dado período de tempo.

4.3.3 O Processo de Monitoração

O processo de monitoração é executado pelo componente do SGME chamado Agente Monitorador. Esse Agente é codificado na linguagem Java, utilizando Classes que implementam recursos para trocas de mensagens SNMPv1 para executar o processo de monitoração. Devido a busca da total capacidade de abstração de hardware e sistemas operacionais, prevista inicialmente, o Agente Monitorador é implementado em linguagem Java.

O Agente Monitorador agrupa, coleta e armazena as informações dos Domínios no banco de dados do SGME. Ele também verifica continuamente a situação em que esses Domínios se encontram, ativando e desativando as suas monitorações quando solicitado pelo Administrador.

4.4 Conclusão

O SGME resulta da articulação das tecnologias de Banco de Dados Relacionais, Cliente/Servidor, Web e de Modelagem de Domínios e Eventos. A sua arquitetura segue o padrão Cliente/Servidor em 3 camadas(*tier*).

A primeira camada provê os meios necessários para se utilizar os serviços do *framework*. Na segunda camada, está o servidor de monitoração que fornece os serviços do SGME. Esses serviços são implementados por um conjunto de *servlets* que são executados pelo servidor *Web* do sistema e por um Agente Monitorador. Já, a terceira camada é composta por agentes SNMP que realizam a

função de servidores de informações de gerenciamento que são armazenadas no servidor de monitoração.

Todos os serviços prestados pelo SGME utilizam um banco de dados cujas definições encontram-se no Modelo de Dados Genérico Dirigido a Eventos - MDGE.

Capítulo 5 - O Modelo de Dados Genérico Orientado a Eventos

5.1 Introdução

O Modelo de Dados Genérico Dirigido a Eventos(MDGE) representa a estrutura de um banco de dados que tem como objetivo apoiar os procedimentos de monitoração de uma rede de computadores via protocolo SNMP. Além disso, permite que os dados oriundos da monitoração sejam facilmente compartilhados com outras aplicações de gerenciamento de redes de computadores.

5.2 As Entidades do MDGE

O MDGE é constituído por um conjunto de Entidades representadas por Tabelas cujos Atributos e Relações definem Quais, Onde, Quando e Como objetos SNMP devem ser monitorados e armazenados. As especificações do MDGE fundamentam-se nas especificações SNMP de [STALLINGS 1999] e nos conceitos de estado, domínio, e eventos apresentados por [HASAN 1999], [PERKINS 1997] e [LIU 1999], respectivamente.

5.2.1 Comunidades

A Entidade Comunidade representa os pré-requisitos para comunicação do SGME e do Agente Monitorador, ambos, com os Agentes SNMP.

A Entidade Comunidade realiza o mapeamento entre os endereços IP dos nodos monitorados, tanto com os nomes das suas respectivas Comunidades como com os seus apelidos, escolhidos pelo Administrador da rede. Eventualmente, tais apelidos são utilizados nas demais Entidades do MDGE como sinônimos dos endereços IP. Essa Entidade tem os seguintes atributos para a realização do mapeamento:

- **ipEqCom** identifica o nodo monitorado. Deve conter o seu endereço IP no formato xxx.xxx.xxx.xxx. O tipo de dado deste atributo é *texto* com tamanho 15. Por exemplo: '200.17.41.166'.
- **nmEqCom** apelida unívoca e sucintamente o nodo monitorado, cujo endereço IP é representado pelo atributo ipEqCom. Um único endereço IP pode estar associado a vários nomes. Deve conter um nome(apelido) para esse nodo. Nesse caso, a primeira palavra ou abreviação do nome utilizado pelo DNS (Domain Name Service) é recomendável. O tipo de dado deste atributo é *texto* com tamanho 25. Por exemplo: 'npds01' ao invés de 'npds01.npd.ufc.br'.
- **nmCom** identifica a Comunidade a qual pertence o nodo monitorado. Deve conter o nome que será utilizado pelo Agente Monitorador para estabelecer a comunicação com os Agentes SNMP dessa comunidade. O tipo de dado deste atributo é *texto* com tamanho 15. Por exemplo: 'public'.

A Figura 5.1 mostra o diagrama da Entidade Comunidade juntamente com a sua visão tabular.

Figura 5.1 - Entidade Comunidade e uma visão, ilustrativa, como tabela

5.2.2 Estados Primitivos

Um objeto é uma entidade que se apresenta num determinado estado e esse estado pode ser modelado como uma coleção de atributos cujos valores podem

mudar no tempo [HASAN 1999]. A partir desse conceito, derivam-se as definições de Estado e Estado Primitivo.

Definição 1 Estado é o valor que uma determinada instância de um objeto gerenciado assume num determinado tempo ***t***. Um Estado pode ser classificado como Primitivo ou Composto.

Definição 2 Cada Estado tomado, em um tempo ***t***, de um nodo da rede é um Estado Primitivo.

Essas definições permitem modelar uma Entidade chamada EstadoPrimitivo que tem os seguintes atributos para um sistema de monitoração:

- **nmEqObjEp** identifica univocamente o nodo monitorado. Deve conter um nome em conformidade com a Entidade Comunidade. O tipo de dado deste atributo é *texto* com tamanho 25. Por exemplo: 'npds01'.
- **idObjEp** identifica univocamente a instância de um objeto gerenciado na MIB do nodo (nmEqObjEp) monitorado. Deve conter o identificador da instância do objeto(OID) alvo da monitoração, formatado em conformidade com especificações SNMP. O tipo de dado deste atributo é *texto* com tamanho 35. Por exemplo: a seqüência '1.3.6.1.2.1.1.1.0' identifica um objeto.
- **nmObjEp** apelida sucintamente o objeto gerenciado representado pelo atributo idObjEp. Deve conter um nome(apelido) para o objeto que será monitorado. Nesse caso, aconselha-se o uso do último nome do caminho da árvore hierárquica que representa a MIB que contém o objeto em questão. O tipo de dado deste atributo é *texto* com tamanho 25. Por exemplo: '*sysDescr*' ao invés *de 'iso.org.dod.internet.mgmt.mib-2.system.sysDescr*' .
- **tpObjEp** Identifica o tipo do objeto gerenciado. Deve ser 'I' caso esse objeto seja do tipo *inteiro,* 'O' caso seja do tipo *octetstring* ou compatível, ou 'C'

caso seja de um dos seguintes tipos: *gauge, counter, timeticks* ou compatível. O tipo de dado deste atributo é *texto* com tamanho 1.

- **fmtObjEp** especifica a forma do objeto gerenciado na MIB. Deve ser 'E' se o objeto for da forma escalar ou 'C' se objeto for da forma colunar. O tipo de dado deste atributo é *texto* com tamanho 1.

A Figura 5.2 mostra a representação gráfica da Entidade EstadoPrimitivo juntamente com um exemplo da sua visão tabular.

Figura 5.2 - Entidade EstadoPrimitvo e uma visão, ilustrativa, como tabela.

O par de atributos <nmEqObjEp, nmObjEp> identifica univocamente cada um dos registros da tabela que implementa a Entidade EstadoPrimitivo. Vários pares desses podem estar associados a um único OID SNMP(idObjEp).

5.2.3 Domínios de Monitoração

Todos os modelos de gerenciamento são fundamentados nas suposições a respeito da utilização do modelo e sobre os seus componentes. A aplicação de um modelo desse tipo a uma situação específica do mundo real é chamada de Domínio de Gerenciamento. Esse termo é uma noção abstrata construída sobre as condições que delimitam, física e administrativamente, as ações de gerenciamento em uma rede [PERKINS 1997]. A partir desse conceito, derivam-se as definições de Domínio e Estado Composto.

Definição 3 O agrupamento de Estados Primitivos forma um Estado Composto que, por analogia com o Domínio de [PERKINS 1997], delimita, física e administrativamente, as ações de monitoração de uma rede de computadores.

Definição 4 Um Domínio identifica um Estado Composto e especifica com que freqüência ele deve ser monitorado. Isso estabelece um regime de observação para um certo Estado Composto. O termo monitoração é utilizado como sinônimo de regime de observação.

O regime de observação pode ser representado pela Entidade Domínio cuja representação gráfica é mostrada na Figura 5.3, juntamente com sua visão tabular.

A Entidade Domínio tem os seguintes atributos:

- **nmDom** identifica univocamente um Domínio de monitoração. Deve conter o nome, arbitrado pelo administrador, do Domínio monitorado. O tipo de dado desse atributo é *texto* com tamanho 25. Por exemplo: 'ufcntFluxIf2'.
- **hrcoletDom** especifica uma data de referência para o cálculo do momento preciso de início do processo de monitoração desse Domínio. Deve conter uma data no formato aaaa/mm/dd hh:mm, onde aaaa, mm, dd, hh, mm representam respectivamente ano, mês, dia, hora e minuto. O tipo de dado desse atributo é *data/hora*. Por exemplo: '2002/08/21 18:31'.
- **freqcoletDom** especifica a freqüência com que o processo de monitoração desse Domínio deve ocorrer. Deve conter um valor inteiro para o tamanho do intervalo de tempo, em minutos, entre cada uma das monitorações(*polling*) desse Domínio. O tipo de dado desse atributo é *inteiro*. Por exemplo: a cada '1' minuto.
- **sitDom** indica a situação em que deve estar o processo de monitoração desse Domínio. Pode assumir os valores 'F' ou 'A' significando, respectivamente, Finalizado e Ativo. O tipo de dado desse atributo é *texto* com tamanho 1. Por exemplo: 'A'.

- **endAgMonDom** especifica o endereço IP do computador encarregado da execução(host) do Agente Monitorador. O tipo de dado desse atributo é *texto* com tamanho 15. Por exemplo: '127.0.0.1'. Nesse exemplo, o Agente Monitorador é executado no mesmo computador que está atuando como Servidor de Monitoração.
- **portAgMonDom** especifica no número da porta TCP através da qual o Agente Monitorador pode trocar mensagens com os outros componentes do SGME para controlar a monitoração desse Domínio. O tipo de dado desse atributo é *inteiro*. Por exemplo: '65000'.

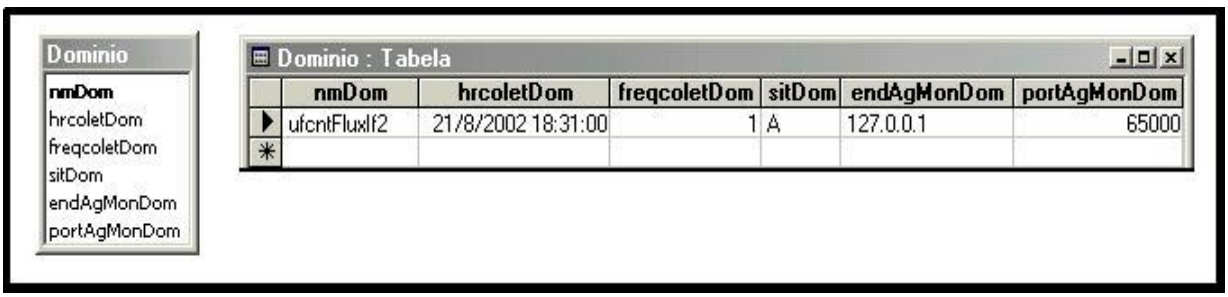

Figura 5.3 - Entidade Domínio e uma visão, ilustrativa, como tabela

Os atributos hrcoletDom e freqcoletDom objetivam permitir uma melhor distribuição, ao longo do tempo, da carga de tráfego de requisições SNMP sobre a rede, imposta pelo processo de monitoração. O cálculo da hora de início de monitoração é dado pela formula:

horacorrente + freqcoletDom - (Resto((MinutosEntre(horacorrente - hrcoletDom) / freqcoletDom))

Para melhor esclarecer a questão do inicio da monitoração de um domínio, suponha que o administrador tenha definido, aproximadamente às 18:32h, que domínio "ufcntFluxIf2" (sitDom indefinido) deve ser monitorado a partir de 18:31h (hrcoletDom), então, de fato, ele só começará a ser monitorado às 18:33, quando terá transcorrido, relativamente à hrcoletDom (18:31), um período de tempo, em minutos, múltiplo da freqüência 1. É nesse momento (18:33) que, de fato, a monitoração do Domínio tem início, conforme o cálculo apresentado abaixo:

18:32 + 1 - (Resto(MinutosEntre(18:32 - 18:31) / 1)) = 18:32 + 1 - (Resto(1 / 1)) = 18:32 + 1 - 0 = 18:33

Apenas objetos com formato escalar ou colunar indexado podem contribuir para a composição de Domínios. Essa restrição visa a simplificação da implementação do processo de monitoração.

5.2.4 Estado Composto

As **Definições 2**, **3** e **4** estabelecem as bases para um relacionamento de associação entre as Entidades Domínio e EstadoPrimitivo. Essa associação indica que um certo Domínio está associado a um ou mais Estados Primitivos, acarretando o surgimento de um Estado Composto. Em outras palavras, determinados subconjuntos de registros da tabela EstadoPrimitivo definem áreas de foco, ou Domínios, para os quais a atenção do Administrador está voltada. Para isso, ele deve utilizar os seus conhecimentos, vivência e criatividade. Essa relação é mostrada pela Figura 5.4, juntamente com a sua visão tabular.

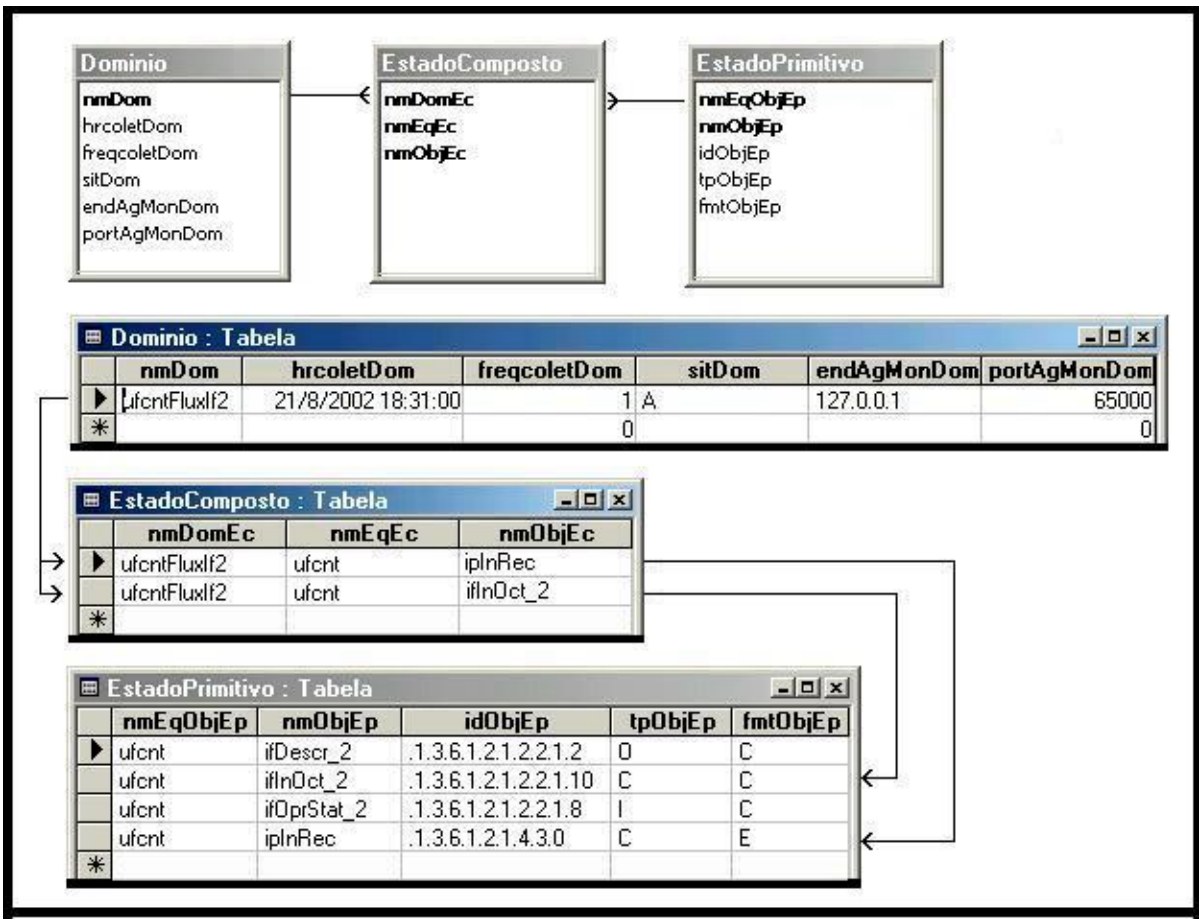

Figura 5.4 - Relacionamento EstadoComposto e uma visão, ilustrativa, das tabelas relacionadas

5.2.5 Índices

Definição 5 Todo objeto SNMP gerenciado de formato escalar é referenciado diretamente pelo seu identificador.

Definição 6 Um Índice é uma seqüência pré-definida de estados que, concatenados numa ordem específica com o identificador de um objeto, permite referenciar uma das suas instâncias enquanto objeto colunar.

Definição 7 Para toda instância de um objeto colunar, existe no mínimo um índice específico.

As **Definições 6** e **7** permitem modelar a Entidade Índice cuja representação gráfica é mostrada na Figura 5.5, juntamente com sua visão tabular.

A Entidade Índice tem os seguintes atributos:

- **nmEqInd** identifica, em conformidade com a Entidade EstadoPrimitivo, o agente SNMP cujo objeto gerenciado precisa ser indexado. O tipo de dado desse atributo é *texto* com tamanho 25. Por exemplo: 'ufcnt'.
- **nmObjEpInd** identifica, em conformidade com a Entidade EstadoPrimitivo, o nome(apelido) do objeto cuja instância precisa ser indexada. O tipo de dado desse atributo é *texto* com tamanho 25. Por exemplo: 'ifDescrIf_2'.
- **ordInd** provê o meio pelo qual os estados representados pelo atributo estadoInd formam um índice. Esse atributo é usado como chave de um processo de classificação ascendente cujo resultado permite que os valores de estadoInd sejam concatenados formando a precisa seqüência indexadora da instância de um objeto colunar. O tipo de dado desse atributo é *inteiro*. Por exemplo: suponha um objeto de forma colunar cuja referência precise de um índice formado por dois valores. Dessa forma, se um estado(estadoInd) tiver ordInd valorado com 7 e o outro valorado com 5, então o estado com ordInd=5 entra em primeiro lugar na seqüência indexadora e o outro, com ordInd=7, entra em segundo lugar.
- **nmObjInd** identifica o objeto cujo estado é utilizado na formação de um Índice. Deve conter o nome desse objeto. O tipo de dado desse atributo é *texto* com tamanho 25. Por exemplo: 'ifNumber'.
- **estadoInd** representa o estado ou valor, propriamente dito, que será concatenado com outros para formar um índice. Deve conter o valor que será usado na formação de um índice por meio de concatenações. O tipo de dado desse atributo é *texto* com tamanho 35. Por exemplo: '2'.

Figura 5.5 - Entidade Índice e uma visão, ilustrativa, como tabela

Agora, com o surgimento da Entidade Índice, e com a criação de um relacionamento associativo entre ela e a Entidade EstadoPrimitivo, o MDGE dá condições ao SGME de referenciar objetos SNMP de ambas as formas, escalar e colunar. A Figura 5.6 mostra essa nova relação com a sua visão tabular.

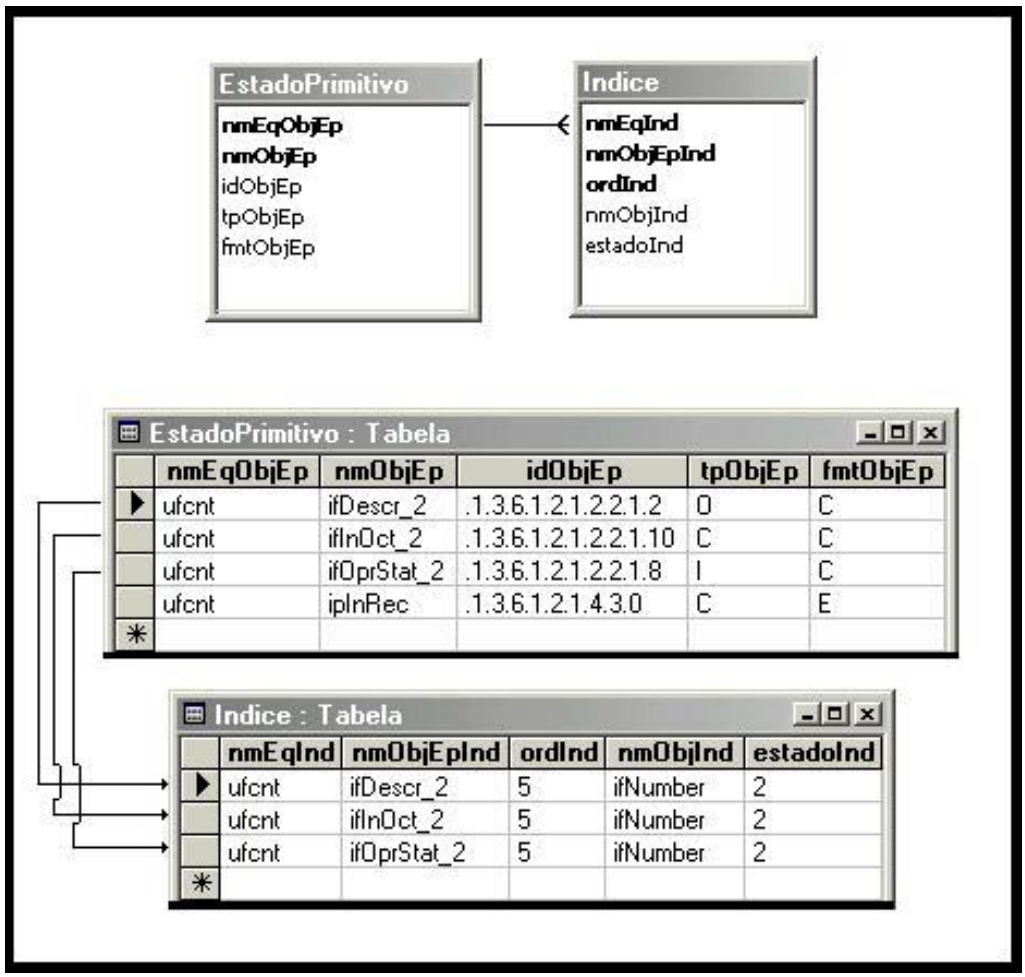

Figura 5.6 - Relacionamento entre EstadoPrimitívo e Índice, e uma visão, ilustrativa, das tabela relacionadas

5.2.6 Repositórios

Definição 8 Para cada Domínio, existe uma estrutura de armazenamento de dados, genericamente, chamada de Repositório. Como um agrupamento de Estados Primitivos define um Estado Composto e este caracteriza um Domínio, então a implementação da Entidade EstadoComposto pode ser vista como uma metatabela cujas informações permitem a construção de outras tabelas. Dessa forma, Repositórios capazes de armazenar os valores dos objetos SNMP monitorados são definidos com base nos Estados Primitivos.

A partir das Entidades que já estão definidas, passam a existir elementos suficientes para a automatização da criação dos Repositórios para os Domínios que podem ser monitorados. O processo de criação das Entidades que modelam cada um dos Repositórios seguirá as seguintes regras:

1 Cada uma dessas entidades são batizadas com o nome do Domínio que representam concatenado com o prefixo "R_";
2 Todo par formado pelos valores dos atributos nmEqObjEp e nmObjEp, obtidos por meio do relacionamento EstadoComposto entre as entidades Domínio e EstadoPrimitivo, dá origem a um atributo cujo tipo será equivalente ao tipo de dado do objeto identificado por nmObjEp em conformidade com valor de tpObjEp;
3 Toda linha de definição desse atributo estará disposta numa ordem formada a partir do agrupamento dos objetos SNMP que compõem um determinado Domínio por ordem de endereço do nome do nodo(nmEqObjEp) e nome do objeto(nmObjEp). O nome de cada um dos atributos do repositório é formado pela concatenação <nmEqObjEp>_<nmObjEp> no caso de objeto com forma escalar e <nmEqObjEp>_<nmObjEp>_<estadoInd>$_1$ ... _<estadoInd>$_n$ no caso de objeto com forma colunar;
4 Cada um desses agrupamentos será precedido por um atributo cujo propósito é datar a coleta dos valores dos objetos SNMP representados pelos atributos criados conforme a regra 3. Seus nomes serão formados pela seguinte concatenação: dtReq_<nmEqObjEp> para nomear o atributo que registra o

momento da requisição dos estados das instâncias dos objetos(idObjEp) ao elemento da rede(nmEqObjEp), e dtResp_<nmEqObjEp> para nomear o atributo que registra o momento do recebimento dos estados requeridos.

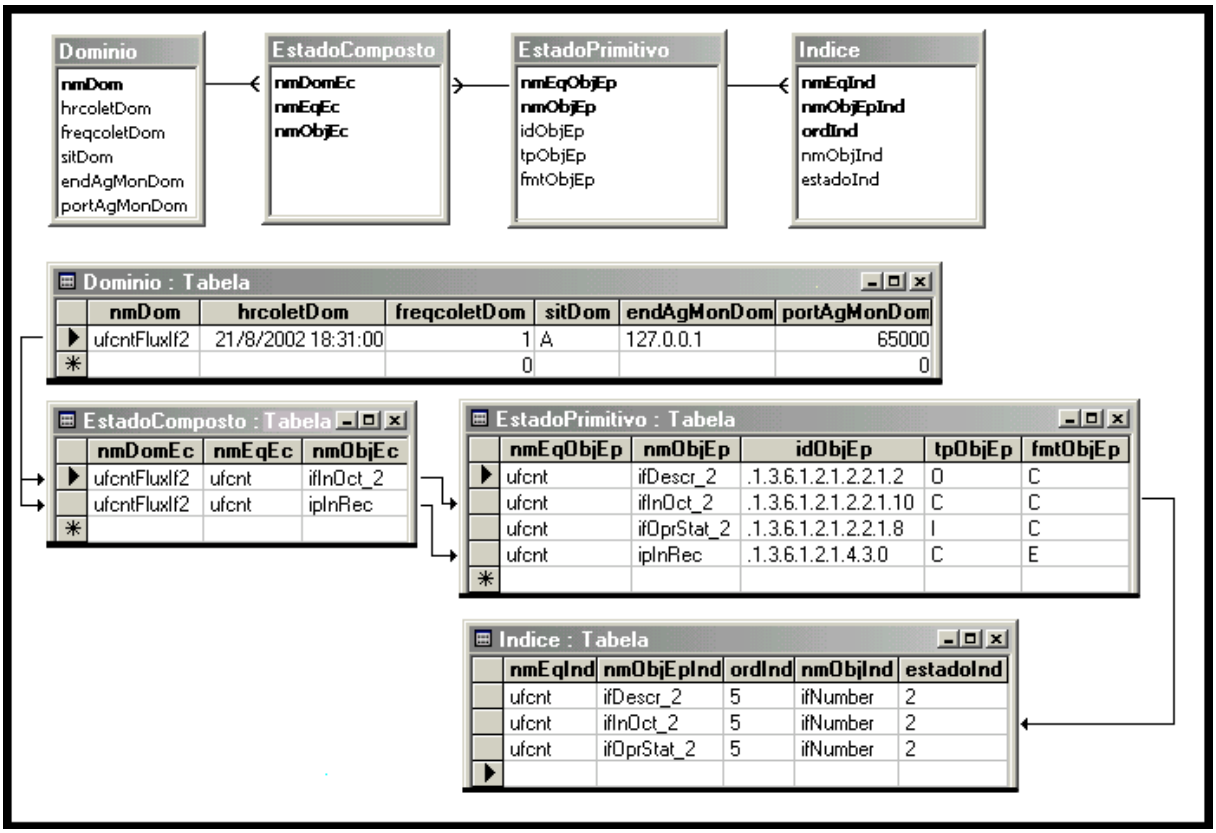

Figura 5.7 - Relacionamentos dos quais se originam as definições dos repositórios

Seguindo essas regras, a Entidade representante do repositório para o Domínio *ufcntFluxIf2*, tendo por base as ilustrações até então mostradas pelas figuras, tem a seguinte definição:

5.2.7 Eventos

Do ponto de vista da engenharia, evento é a alteração, administrativamente significante, de um estado que pode ocorrer em um sistema. Um evento primitivo é uma mudança de estado pré-definida em um sistema e o seu mecanismo de detecção, normalmente, está embutido nesse mesmo sistema. Um evento composto é formado pela composição de eventos primitivos e/ou de outros eventos compostos, [LIU 1999]. A partir desse conceito, derivam-se as definições de Restrição e Evento Simples, Fórmula e Evento Composto.

Definição 9 Restrição é o confinamento de um Estado Primitivo a um determinado estado ou a uma faixa de variação de estados. Quando um Estado Primitivo viola a sua Restrição, o resultado da sua avaliação lógica é Verdadeiro, caso contrário é Falso.

Definição 10 Um Evento Simples ocorre quando um Estado Primitivo não satisfaz a sua restrição, isto é, a avaliação lógica dessa Restrição resulta Verdadeiro.

A partir das **Definições 9** e **10**, torna-se possível expandir a Entidade EstadoPrimitivo para que ela comporte os conceitos de Restrição e Evento Simples.

Assim, a Entidade EstadoPrimitivo passa a conter adicionalmente os seguintes atributos:

- **restrEp** indica se uma restrição deve ser verificada caso esse Estado Primitivo componha a definição de algum evento. Deve ser "S" indicando que esse Estado Primitivo está sujeito a alguma restrição, ou "N" caso contrário. O tipo de dado desse atributo é *texto* com tamanho 1. Os atributos vlMaxEp e vlMinEp, a seguir, especificam os limites dessa restrição.

- **vlMaxEp** define o limite superior da restrição. Deve conter o maior valor que uma instância de um objeto dessa Entidade pode assumir. O tipo de dado desse atributo é *texto* com tamanho 25.
- **vlMinEp** define o limite inferior da restrição. Deve conter o menor valor que uma instância de um objeto dessa Classe pode assumir. O tipo de dado desse atributo é *texto* com tamanho 25.

Se os valores máximo(vlMaxEp) e mínimo(vlMinEp) são iguais, significa que o Estado Primitivo deve assumir precisamente esse valor para não representar um Evento Simples. Vale ressaltar, também, que embora esse valores sejam armazenados como texto, eles são interpretados, de fato, em conformidade com a tabela 5.1 de conversão de tipos de dados dos Estados Primitivos:

Tabela 5.1 - Conversão de tipos de dados

Tipo SNMP	Tipo JDBC	Tipo Java
'O' - octetstring ou compatível	string	string
'I' - integer	double	double
'C' - counter, gauge, timeticks ou compatível	double	double
-	date	java.sql.Date

A Figura 5.8, a seguir, representa a expansão da Entidade EstadoPrimitivo cujas associações com as Classes Domínio e Índice permanecem inalteradas. Essa figura, também, mostra a nova visão tabular dessa Classe.

Figura 5.8 - Entidade EstadoPrimítivo expandida e uma ilustração da sua nova forma tabular

Definição 11 Fórmula é uma expressão lógica na qual Restrições são operandos e os símbolos ^(e), v(ou) e ~(não) são operadores. Uma Restrição se traduz pelo par <nmEqObjEp, nmObjEp> que identifica um Estado Primitivo cujo atributo restrEp está valorado com "S". Uma fórmula representa um Evento Composto.

Definição 12 Um Evento Composto ocorre quando a avaliação lógica de uma Fórmula resulta Verdadeiro.

As definições 11 e 12 lançam as bases para as especificações da Entidade Fórmula. Elas modelam as condições que representam os Eventos Compostos que, doravante, passam a ser referenciados apenas como Eventos.

A Entidade Fórmula tem os seguintes atributos:

- **nmEvCompF** identifica univocamente um Evento. Deve conter o nome do Evento Composto representado pela equação armazenada em expLogEvCompF. O tipo de dado desse atributo é *texto* com tamanho 25.
- **nmDomF** indica, em conformidade com a Entidade Domínio, o Domínio onde o Evento identificado pelo atributo nmEvCompF pode ocorrer. O tipo de dado desse atributo é *texto* com tamanho 25.
- **descrEvCompF** descreve um Evento Composto. O tipo de dado desse atributo é *texto* com tamanho livre.
- **reaçãoEvCompF** descreve a reação que o Administrador deve tomar com relação a ocorrência do Evento em questão. O tipo de dado desse atributo é *texto* com tamanho livre.
- **expLogEvCompF** expressa a sintaxe da sentença SQL construtora da Visão, ou *View*, condicionada por uma fórmula em conformidade com a definição 11. O tipo de dado desse atributo é *texto* com tamanho livre.

A Figura-5.9 mostra a Entidade Fórmula juntamente com a sua relação de associação com a Entidade Domínio e a sua visão tabular.

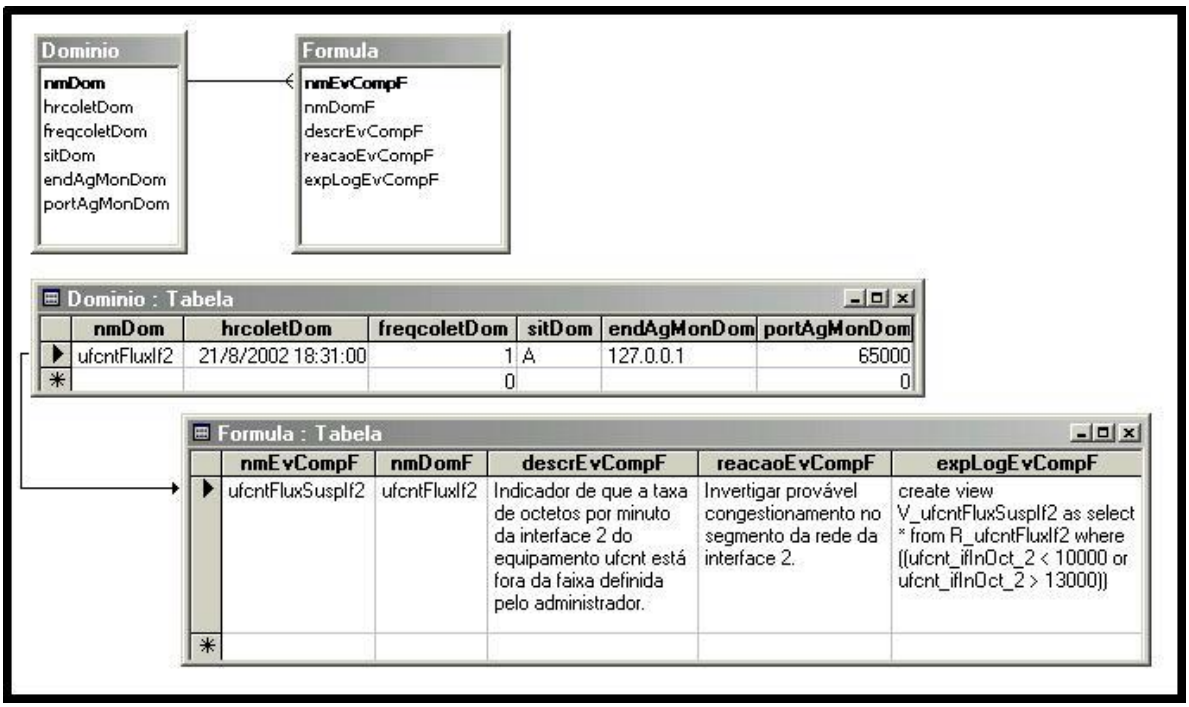

Figura 5.9 - Relacionamento entre as Entidades Fórmula e Domínio

Aqui é importante perceber que a Entidade Fórmula tem caráter, em geral, apenas documentacional. Portanto, faz-se necessário criar o mecanismo que efetivamente detecte a ocorrência dos Eventos definidos. Dentro dessa proposta, tal mecanismo é obtido de modo direto, simples e natural por meio de visões ou *views* do SGBDR.

As visões são definidas por meio do emprego da Fórmula que define um Evento Composto, na construção da uma sentença *sql*, como uma regra de seleção aplicada sobre o Repositório que armazena os dados de um determinado Domínio.

Essa regra ou mecanismo é acionado no preciso momento em que se grava dados no Repositório. Assim, a detecção do evento é imediata, tornando o sistema apto tanto a produzir relatos históricos sobre as ocorrências, como a sugerir algum procedimento de reação técnico-administrativo para os operadores da rede.

A automação das reações pode ser obtida pela utilização de *Triggers* e *Stored Procedures* juntamente com as Visões, mas isso foge ao escopo desse trabalho.

5.3 Conclusão

O Modelo de Dados Genérico Dirigido a Eventos incorpora, na forma de estrutura de dados, os conceitos de Estados, Domínios de Gerenciamento e Eventos. Ele dá suporte à parametrização do processo de monitoração e ao armazenamento de informações gerenciais e de geração de relatórios do SGME. Além disso, provê um mecanismo de detecção on-line de Eventos. A partir das suas especificações, é possível definir e implementar um protótipo cujas interfaces e funções permitam alcançar os objetivos do SGME.

Capítulo 6 - A Implementação do SGME

6.1 Introdução

O projeto arquitetônico dos componentes do SGME, juntamente com a modelagem dos objetos SNMP em um banco de dados relacional, compõem uma estrutura que pode ser utilizada para apoiar a construção das possíveis aplicações de gerenciamento de redes de computadores baseados no protocolo SNMP.

A obtenção do protótipo do SGME está baseada nessa estrutura e é alcançada por meio de um conjunto de pequenos programas de computadores que produzem serviços, cujas requisições fazem com que os componentes interajam, trocando dados entre si apoiados por um banco de dados relacional. Esse conjunto de programas pode ser separado em três grupos de acordo com os serviços que eles provêem.

No primeiro grupo, estão os serviços que disponibilizam os meios necessários para a monitoração realizada pelo terceiro grupo. O segundo grupo engloba os serviços que possibilitam a definição e a geração automáticas de alguns tipos de relatórios gerenciais com base nos dados coletados. O terceiro grupo é formado apenas pelo Agente Monitorador cuja função é executar o processo de monitoração.

Em virtude da busca da operacionalidade do nosso *framework* em quaisquer tipos de plataformas, e ainda mais, através da Internet, todos os seus serviços são realizados por funções codificadas na linguagem Java, sendo que os dois primeiros grupos são formados por mini-aplicativos, normalmente, conhecidos como Java servlets doravante chamados de Transações.

Esse capítulo apresenta a implementação das funções do SGME.

6.2 A Modelagem Visual do SGME

A interface do SGME é totalmente baseada nos padrões de apresentação e navegação em conteúdos da Internet. Como é sabido, nesse contexto, os conceitos *apresentar* e *navegar* trazem implícito a necessidade de trocas de informações. Assim, a utilização do protocolo HTTP e da linguagem HTML como ferramentas para a implementação das funções foi uma conseqüência natural. O protocolo HTTP provê a comunicação de dados entre as primeira e segunda camadas do nosso modelo arquitetônico, e a linguagem HTML permite a codificação das apresentações das interfaces que modelam graficamente os dados manipulados pelo *framework*.

No campo da apresentação, a implementação das interfaces foi feita da maneira mais simples possível. Um repertório de fontes bastante reduzido, quase sem variação de cores, tipos e tamanhos, foi utilizado buscando obter telas visualmente confortáveis. Numa boa parte das interfaces, os dados são alimentados por meio de simples "cliques" do *mouse*. Isso reduz a utilização do teclado que em certas circunstâncias é menos amigável.

Todos os objetos das interfaces estão dispostos em conformidade com os hábitos ocidentais de leitura. Eles estão colocados de cima para baixo e da esquerda para direita. Essa disposição, também, retrata o fluxo natural com que as transações devem ser operadas. No SGME, existe uma interface gráfica primária fixa que é permanentemente apresentada. Ela está dividida em três áreas conforme mostra a Figura 6.1.

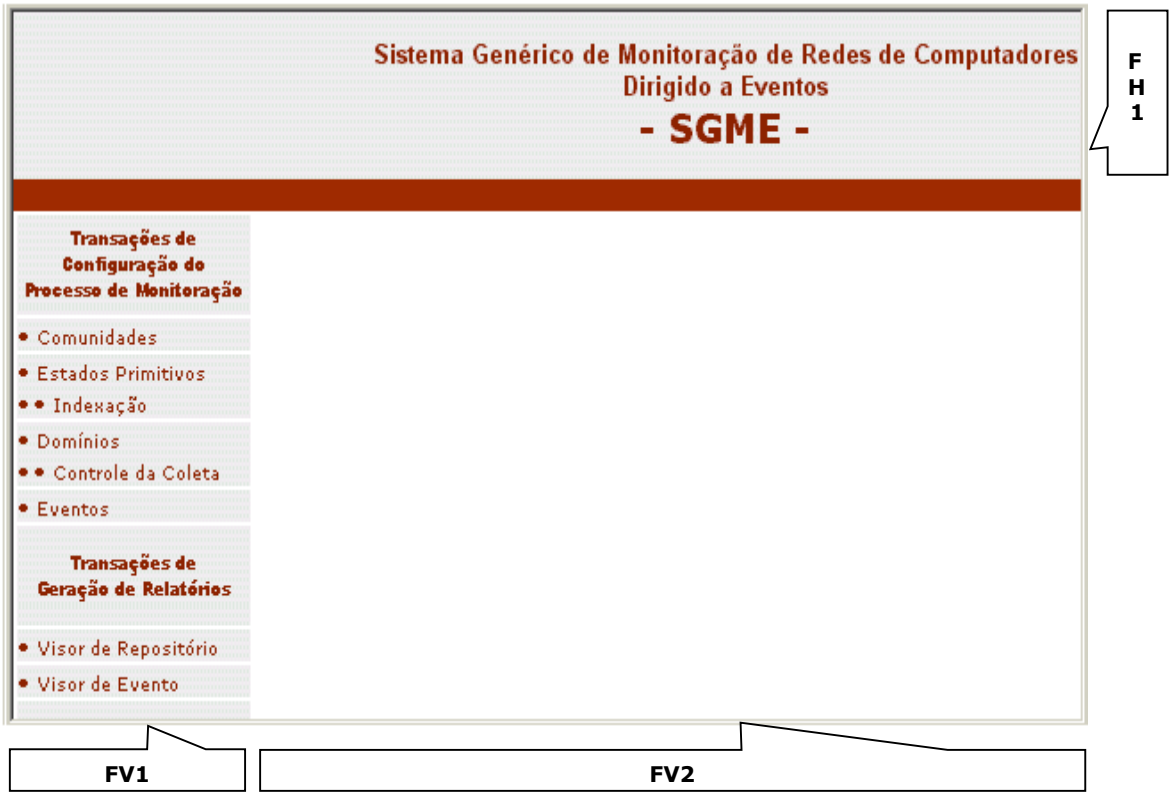

Figura 6.1 - Interface primária. Disposição dos *frames*

A apresentação dessa interface primária é caracterizada pela divisão da área de trabalho do *Browser*, inicialmente, em dois *frames* horizontais. A parte superior, o frame FH1, é apenas uma pequena porção horizontal utilizada para entitular o ambiente onde o usuário está inserido. A parte inferior, por sua vez, é subdividida em dois outros *frames* verticais. A parte vertical mais à esquerda, o *frame* FV1, é o menu de transações do SGME. A parte restante à direita é o *frame* FV2 que corresponde à área de trabalho do SGME. É nele que as transações do *framework* apresentam as suas interfaces.

Numa tentativa de deixar transparecer a hierarquização das dependências funcionais entre as transações, elas foram dispostas verticalmente no *frame* de menu. O objetivo dessa disposição é informar visualmente que o funcionamento das transações situadas na parte mais baixa do menu dependem da utilização das que estão imediatamente mais acima.

Por fim, no campo da visualização, nossas interfaces são projetadas para monitores de 15 polegadas com resolução de 800 por 600 *pixels*, por serem estas as configurações disponíveis no ambiente de desenvolvimento do protótipo do SGME.

No campo da navegação pelas transações, foram tomados cuidados com relação à navegação visual dentro das interfaces e com relação à navegação operacional entre as interfaces.

Com relação à navegação visual dentro das interfaces, optou-se pela utilização de objetos de entrada de dados e de acionamento de ações que mais se adequassem às convenções utilizadas na Internet. Desse modo, espera-se que caixas de entradas, listas de seleção, botões do tipo Avançar, Excluir, Fim, etc., além dos *links* de navegação, por si só, explicitem as suas utilidades. A adoção desses objetos em conjunto com etiquetas e textos sucintos busca proporcionar uma compreensão quase intuitiva do funcionamento das interfaces e conseqüentemente das transações do *framework*. As interfaces buscam evitar a ocorrência de mecanismos de *scrolling*.

Com relação à navegação operacional entre as interfaces, foram padronizados alguns *layouts* que se repetem ao longo da utilização das transações. Dessa forma, a visualização e utilização gradativa das interfaces tornam a operação do SGME amigável. As interfaces estão implementadas de forma a proporcionar uma navegação em que o usuário não precisa passar, em geral, por mais do que duas ou três telas para chegar ao ponto no qual se encontra a função precípua da transação em uso. Com isso, simplifica-se a navegação dentro das transações(entre interfaces).

E a navegação entre as transações está ainda mais simplificada pela presença constante no menu do SGME na tela do *browser*. Complementarmente, todas as interfaces apresentam um cabeçalho que indica a transação que está em

curso e, também, quando for o caso, os dados oriundos da interface anterior à que está sendo apresentada.

Essa modelagem visual objetiva facilitar a compreensão dos serviços prestados pelo *framework* implementado.

6.3 O Funcionamento do SGME

O funcionamento geral do SGME consiste em:

- Configurar o processo de monitoração;
- Definir e gerar relatórios gerenciais;
- Monitorar Domínios.

6.3.1 A Configuração do Processo de Monitoração

A configuração do processo de monitoração é feito por meio das seguintes transações: Registro de Comunidades, Registro de Estados Primitivos, Registro de Indexação, Registro de Domínios, Registro de Controle da Coleta e Registro de Eventos.

6.3.1.1 Registro de Comunidades

A Transação Registro de Comunidades provê as funções de manutenção da tabela que implementa a Entidade Comunidade do MDGE. Seu objetivo é identificar todos os equipamentos da rede que podem ser alvos de algum processo de monitoração. Ela permite que tais equipamentos sejam referenciados, posteriormente, dentro do *framework* de modo mais mnemônico por meio de um nome(apelido) sucinto. O apelido é associado a um endereço IP e a um nome de comunidade SNMP. Essa transação apresenta duas interfaces gráficas.

A função da primeira interface é apresentar todas as comunidades já registradas por essa mesma transação, abrindo a possibilidade de criação de uma nova comunidade. A segunda interface tem como função a manutenção da

comunidade identificada na primeira interface. A manutenção consiste em registrar uma nova comunidade ou em manipular o registro dos equipamentos de uma já existente, dependendo dos dados alimentados através da primeira interface. Em ambos os casos, essa manutenção é feita por meio de inclusões e exclusões de equipamentos na tabela Comunidade. Cada equipamento é identificado pelo seu endereço IP e por um nome.

A Figura 6.2 exibe através das setas a navegação típica entre as interfaces dessa transação. Ela exemplifica o registro de uma comunidade chamada "public" composta por apenas um equipamento.

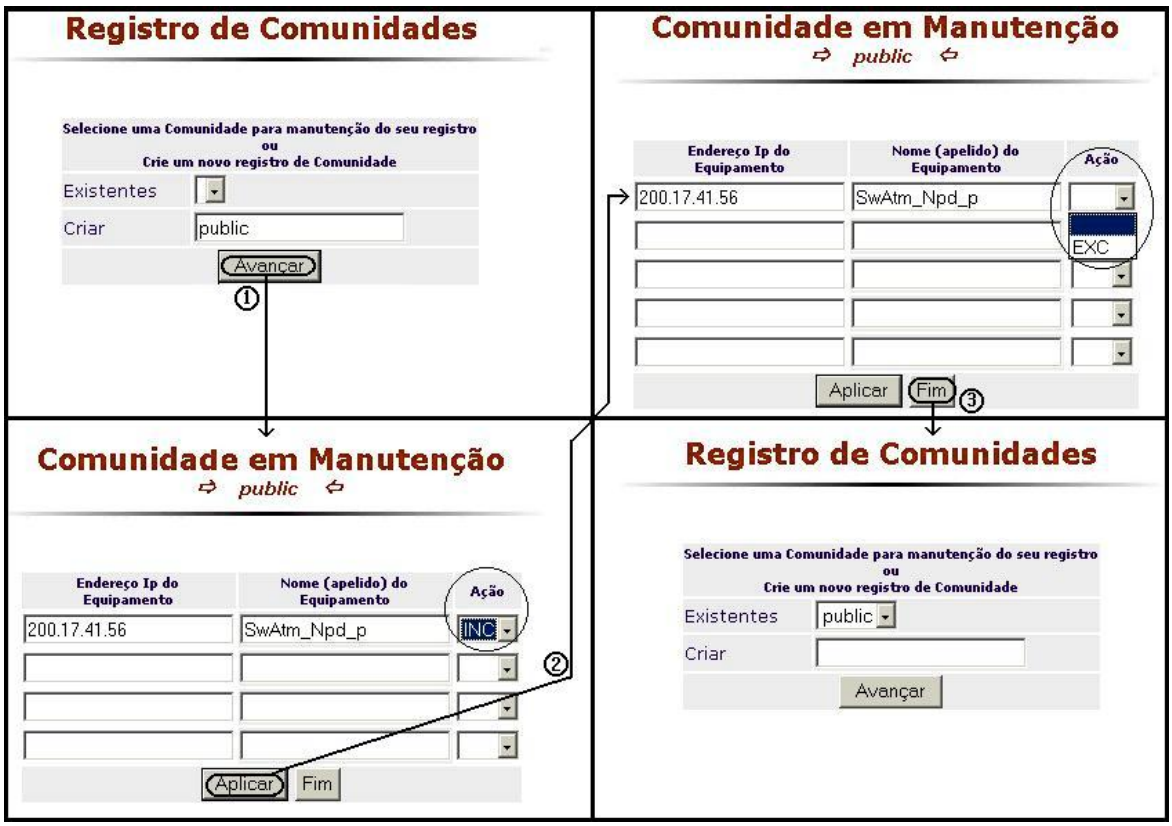

Figura 6.2 - Fluxo típico da inclusão de uma comunidade no SGME

Na primeira interface, identifica-se a nova Comunidade a ser criada, "public", ou, em caso de manutenção, seleciona-se uma Comunidade dentre aquelas apresentadas como já existentes. Em ambos os casos, a transição para a

segunda interface ocorre com o acionamento do botão Avançar, conforme mostra a seta 1. A segunda interface apresenta um formulário onde são especificados os equipamentos pertencentes à Comunidade. Nesse exemplo, ela é composta pelo computador cujos endereço IP e nome são respectivamente "200.17.41.56" e "SwAtmNpd_p". Para que esses dados sejam armazenados no banco de dados, em cada novo equipamento especificado, a opção INC da lista de seleção Ação deve ser escolhida. Ao final do preenchimento do formulário, o botão Aplicar deve ser acionado. Com isso, a transação grava os dados do formulário na tabela Comunidade e exibe imediatamente de volta os componentes registrados conforme a transição mostrada pela seta 2. Nesse novo contexto, a lista de seleção de Ação para equipamento já registrado só oferece a opção EXC para exclusão de componente da Comunidade. A execução das Ações INC e EXC só é efetivada por meio do acionamento do botão Aplicar.

Esse tipo de interface para manipulação dos registros das tabelas, além dos conteúdos que normalmente exibe, sempre se apresenta contendo quatro linhas de campos vazios para alimentação de mais informações. Esse recurso permite a alimentação de tantas informações quantas forem necessárias. Doravante, a função da lista de seleção Ação não mais será comentada por se tratar de um mecanismo padronizado dentro do SGME.

O acionamento do botão Fim faz a transação retornar para a primeira interface conforme mostra a seta 3. Nesse caso, a Comunidade "public" passa a constar na lista de seleção de comunidades existentes como resultado do acionamento do botão Aplicar.

6.3.1.2 Registro de Estados Primitivos

A Transação Registro de Estados Primitivos provê as funções de manutenção da tabela que implementa a Entidade EstadoPrimitivo do MDGE. Seu objetivo é definir objetos de um Agente SNMP como Estados Primitivos e verificar o estado operacional desse Agente. A definição de um Estado Primitivo é feita

pela associação do OID de um certo objeto a um nome(apelido) sucinto, mas mnemônico, que é utilizado dentro do próprio *framework*. Complementarmente, são especificados o tipo de dado, a forma e a Restrição desse objeto. A verificação do estado operacional de um Agente SNMP é feita por meio da geração automática da sua Ficha Técnica. Essa transação apresenta três interfaces.

A função da primeira interface é apresentar todos os equipamentos, ou Agentes SNMP, mantidos pela transação Registro de Comunidades, abrindo a possibilidade para registro de Estados Primitivos e para a verificação do estado operacional desses equipamentos. A função da segunda interface é fazer a manutenção dos Estados Primitivos dos Agentes SNMP cujos nomes são apresentados na interface anterior. Essa manutenção é feita por meio de inclusões e exclusões de objetos SNMP com seus respectivos provedores e demais atributos na tabela EstadoPrimitivo. A terceira interface tem como função a criação de uma Ficha Técnica contendo os dados da configuração e do estado operacional dos equipamentos cujos endereços IP são apresentados pela primeira interface.

A Figura 6.3a, exibe através das setas, a navegação típica entre a primeira e a segunda interface dessa transação. Ela exemplifica o registro de quatro Estados Primitivos(Objetos SNMP) do Agente SNMP (equipamento) chamado "ufcnt".

Figura 6.3a- Fluxo típico da inclusão de Estados Primtivos no SGME

A primeira interface apresenta uma relação de pares de informações formados por um nome de um Agente SNMP e pelo seu endereço IP. Ambas as informações são *links* que permitem navegar entre as demais interfaces da transação.

O ato de apontar e 'clicar' sobre o nome de um Agente causa a transição para a segunda interface como mostra a seta 1. Ela apresenta um formulário onde

são especificados os registros dos Estado Primitivos que o Agente "ufcnt" pode prover. No exemplo, os OIDs, nomes, tipos, formas e Restrições dos objetos que caracterizam cada Estado Primitivo são precisamente os que estão apresentados na Figura 6.3a. A gravação desses registros no banco de dados requer a escolha da opção INC das listas de seleção Ação de cada registro de definição de Estado Primitivo. O acionamento do botão Aplicar resulta na gravação dos dados do formulário na tabela EstadoPrimitivo e apresenta, novamente, todos os registros relativos ao equipamento "ufcnt" conforme a transição mostrada pela seta 2. A exclusão de registros segue o mesmo mecanismo associado à lista de seleção de Ação que foi explicado no Registro de Comunidades.

A Figura 6.3b exibe através das setas a navegação típica entre a primeira e a terceira interface dessa transação. Ela exemplifica a apresentação do estado operacional do equipamento "ufcnt".

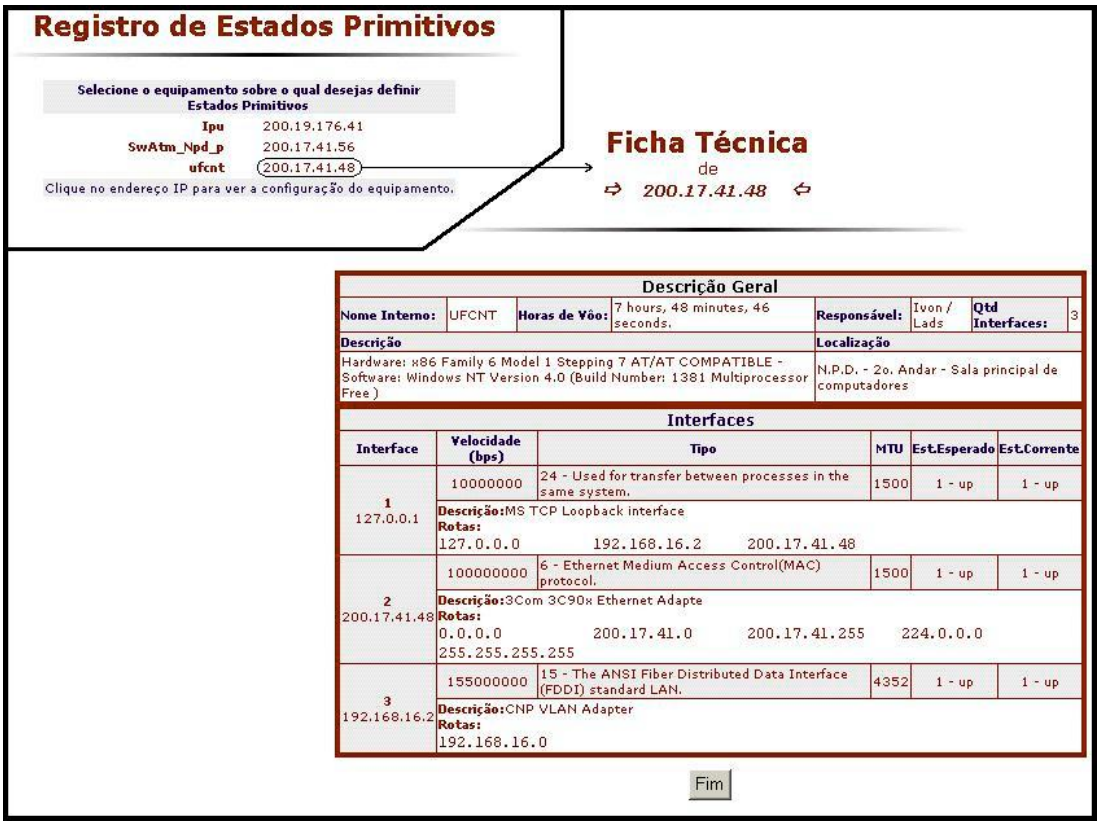

Figura 6.3b - Fluxo típico para obtenção da Ficha Técnica de um nodo da rede

O ato de apontar e "clicar" sobre o endereço IP "200.17.41.48" do equipamento "ufcnt" causa a transição da primeira para a terceira interface como mostra a seta 1. A terceira interface solicita algumas informações ao equipamento apontado para produzir uma Ficha Técnica a seu respeito. Nela são exibidas as seguintes informações: nome interno do equipamento, tempo de funcionamento ininterrupto do equipamento, nome do responsável pelo equipamento, quantidade de interfaces do equipamento, descrição do equipamento, localização do equipamento, identificador de cada interface com seus respectivos endereço IP, velocidade, tipo, MTU(largest protocol data unit), estado desejado e corrente, descrição e rotas que passam pela interface. Todas essas informações são obtidas diretamente por meio do protocolo SNMP. Caso alguma falha ocorra impedindo a resposta a essa requisição, uma interface que relaciona as possíveis causas é então apresentada juntamente com o erro de comunicação ocorrido propriamente dito, como é mostrado na Figura 6.3c.

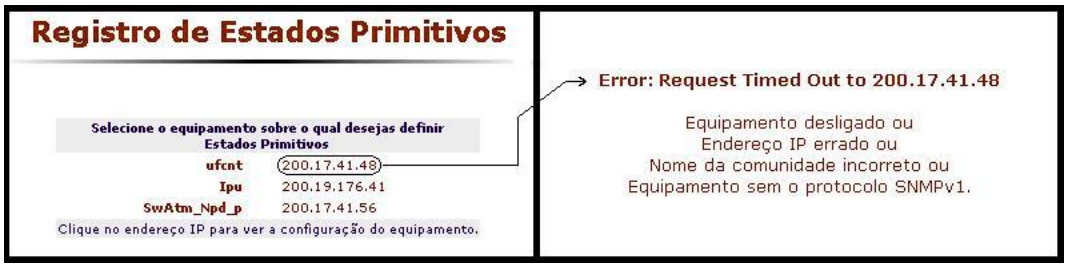

Figura 6.3c - Possíveis falhas de comunicação com um equipamento da rede

6.3.1.3 Registro de Indexação

A Transação Registro de Indexação complementa as funções do Registro de Estados Primitivos que, isoladamente, permite ao SGME referenciar apenas objetos escalares. Essa transação provê as funções de manutenção da tabela que implementa a Entidade Índice cujo objetivo é dar condições ao *framework* proposto de referenciar também objetos colunares. A Transação Registro de Indexação apresenta três interfaces.

A função da primeira interface é apresentar todos os Estados Primitivos definidos a partir de objetos colunares já registrados, abrindo a possibilidade para registro dos seus indexadores e para a verificação da especificação dos seus OIDs. A segunda interface tem como função fazer a indexação ou registrar os indexadores de um objeto colunar. E a função da terceira interface é apresentar todas as instâncias dos objetos cujos OIDs estão hierarquicamente abaixo do OID do objeto que se deseja verificar na árvore da MIB. Em outras palavras, a terceira interface faz uma caminhada(*walk*) a partir do OID não indexado de um objeto colunar na sua respectiva MIB.

A Figura 6.4a ilustra através das setas a navegação entre a primeira e segunda interface. Ela exemplifica o registro da indexação do Estado Primitivo chamado "ifDescr_2".

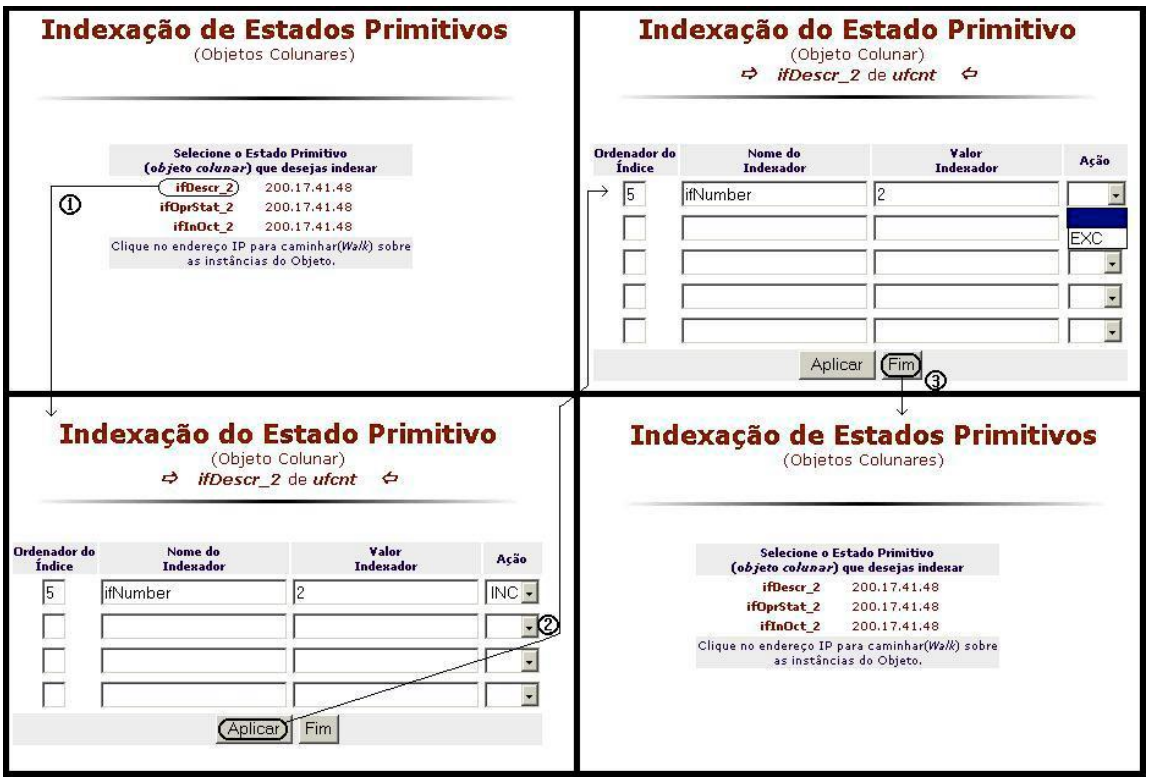

Figura 6.4a - Fluxo típico de indexação de Estados Primtivos no SGME

A primeira interface apresenta uma relação de Estados Primitivos representados por objetos colunares, possivelmente não indexados, juntos aos endereços IP dos seus respectivos provedores. O ato de apontar e "clicar" sobre um Estado Primitivo causa a transição para a segunda interface como mostra a seta 1. A segunda interface apresenta um formulário onde são especificados os indexadores do objeto que representa o Estado Primitivo apontado na primeira interface. No caso exemplificado, de acordo com as especificações SNMP do objeto em questão, apenas um indexador é necessário para compor o seu índice. Nesse contexto, ele é representado pelo objeto "ifNumber" com o valor "2". Desse modo, o OID usado na especificação do Estado Primitivo em questão é dado por ".1.3.6.1.2.1.2.2.1.2" concatenado com ".2".

A formação de OIDs indexados requer que a concatenação dos indexadores ocorra conforme as especificações estabelecidas pelo protocolo SNMP para cada objeto colunar. Portanto, essa interface oferece um recurso chamado Ordenador do Índice. Ele permite que a concatenação ocorra em função de pesos atribuídos para cada um dos indexadores. Os indexadores com menores pesos são os primeiros concatenados e os com maiores pesos são concatenados por último.

A inclusão ou exclusão dos indexadores na tabela Índice requer a escolha das Ações INC ou EXC. O acionamento do botão Aplicar resulta na execução dessas ações com a imediata apresentação de todos os indexadores que formam o índice do objeto colunar conforme a transição mostrada pela seta 2 da Figura 6.4a.

A Figura 6.4b exibe através das setas a navegação entre a primeira e a terceira interface. Ela exemplifica a apresentação de uma caminhada sobre a MIB do Agente SNMP cujo endereço IP é "200.17.41.48" a partir do Estado Primitivo representado pelo objeto parcialmente identificado pelo OID ".1.3.6.1.2.1.2.2.1.2".

Figura 6.4b - Ilustração de uma caminhada na MIB-2

O ato de apontar e 'clicar' sobre o endereço "200.17.41.48" causa a transição para a terceira interface. Ela realiza uma caminhada(*walk*) sobre todas as instâncias do objeto colunar e apresenta todos os seus OIDs indexados com os respectivos tipos e valores. Com isso, é possível verificar, visualmente, se uma dada instância é de fato a que se deseja monitorar e se o seu OID foi corretamente registrado. A seta 1 mostra essa transição.

O acionamento do botão Fim sempre ocasiona a transição para a primeira interface, conforme ilustram as figuras 6.4a e 6.4b.

Como resultado das Transações Registro de Comunidades, Registro de Estados Primitivos e Registro de Indexação, o *framework* SGME passa a permitir a manipulação dos objetos SNMP de modo mais intuitivo por meio dos nomes dos Estados Primitivos. Esses nomes trazem, implicitamente consigo, os OIDS do objetos, devidamente indexados quando necessário, as suas Restrições, os nomes e endereços IP dos seus provedores. Além disso, os Estados Primitivos podem ser referenciados por meio de simples "cliques" sobre seus nomes ou sobre caixas de seleções a eles associadas.

6.3.1.4 Registro de Domínios

A Transação Registro de Domínios provê as funções de manutenção das tabelas que implementam as Entidades Domínio e EstadoComposto. O seu objetivo é agrupar os Estados Primitivos em Estados Compostos formando Domínios de monitoração, isto é, grupos de objetos que na concepção do administrador da rede estão sujeitos a algum tipo de relacionamento lógico quando seus estados ou valores são todos coletados em um mesmo instante do tempo. Todos esses objetos podem ser providos tanto por um mesmo Agente SNMP como por Agentes SNMP distintos. Essa transação apresenta duas interfaces.

A função da primeira interface é apresentar todos os Domínios registrados, porém inativos, abrindo a possibilidade para a criação de um novo Domínio. A segunda interface tem como função a manutenção dos Domínios. A manutenção consiste em registrar um novo Domínio ou em manipular os registros dos componentes de um Domínio já existente. Em ambos os casos, isso é feito por meio de inclusões e exclusões de Estados Primitivos na tabela EstadoComposto.

A Figura 6.5 exibe através das setas a navegação típica entre as interfaces dessa transação. Ela exemplifica o registro de um novo Domínio chamado "ufcntFluxIf2" formado por Estados Primitivos providos pelo Agente SNMP "ufcnt".

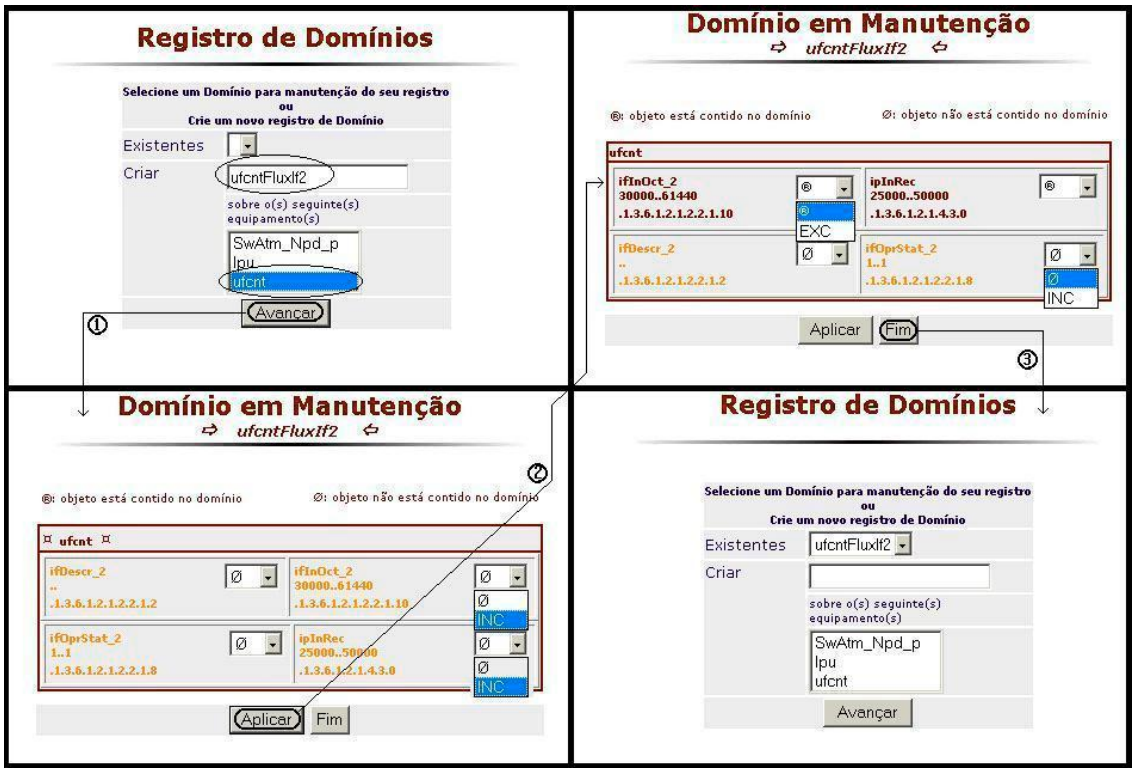

Figura 6.5 - Criação de um Domínio de Monitoração

Na primeira interface, no caso de criação, especifica-se o nome do Domínio, "ufcntFluxIf2", juntamente com os equipamentos(Agentes SNMP) que devem prover os Estados Primitivos para a sua composição, nesse exemplo, o Agente "ufcnt". Isso por si só já representa a etapa inicial do processo de agrupamento dos componentes de um Domínio. Em caso de manutenção, é necessário apenas a seleção de um Domínio dentre aqueles apresentados como já existentes. O SGME não permite a manutenção de Domínios ativos devido às prováveis mudanças de *layout* dos seus repositórios. Em ambos os casos, a transição para a segunda interface ocorre com o acionamento do botão Avançar, conforme mostra a seta 1.

A segunda interface apresenta o nome do Domínio que está em manutenção ("ufcntFluxIf2"), e os equipamentos("ufcnt") previamente selecionados, emoldurados com seus respectivos Estados Primitivos que podem ser escolhidos a critério do Administrador da rede para compor um Domínio. Essas escolhas são

feitas no mesmo estilo como se utilizam as caixas de Ação já apresentadas. Nessa interface, os Estados Primitivos que não compõem um Domínio apresentam o símbolo vazio(∅) como primeira opção da caixa de Ação e os que fazem parte apresentam o símbolo registrado(®). Além disso, os nomes, as Restrições, e OIDs dos objetos representados por Estados Primitivos apresentam-se com uma cor diferenciada, mais escura na interface. A inserção ou remoção de objetos em um Domínio é feita por meio da escolha das opções INC ou EXC nas suas respectivas caixas de Ação. O acionamento do botão Aplicar causa a execução dessas ações sobre as tabelas Domínio e EstadoComposto com imediata apresentação da composição atual do Domínio conforme mostra a seta 2. Dessa forma é possível fazer inserções e exclusões de componentes em um Domínio conforme for necessário.

O acionamento do botão Fim faz a transação apresentar a primeira interface conforme indica a seta 3. Nela o domínio "ufcntFluxIf2" passa a constar na relação de Domínios existentes.

6.3.1.5 Registro de Controle da Coleta

A Transação Registro de Controle da Coleta provê as funções de verificação das especificações que definem um Domínio e de controle dos seus processos de monitoração. Tem como objetivos permitir: a verificação funcional dos Domínios; a verificação do registro dos Domínios; a inicialização e a finalização da monitoração dos Domínios. Essa transação apresenta cinco interfaces.

A função da primeira interface é apresentar todos os Domínios que estão sob controle do SGME. A segunda interface tem como função a verificação funcional dos Domínios por meio da simulação dos *pollings* que devem ser realizados pelos Agentes Monitoradores. A terceira interface apenas apresenta as especificações de um Domínio para simples verificação visual. As funções da quarta e da quinta interface são, respectivamente, inicializar e finalizar o processo de monitoração de um certo Domínio.

A Figura 6.6a exibe a navegação entre a primeira e a segunda interface, e entre a primeira e a terceira interface. Ela exemplifica a verificação funcional e a verificação do registro do Domínio “ufcntFluxIf2”.

A primeira interface exibe uma relação de todos os domínios em fase de definição ou com a definição concluída. Cada linha dessa relação é formada pelo ícone Binóculo e pelo nome de um Domínio seguidos de um indicador da situação funcional desse domínio e de uma Ação.

O ícone Binóculo é um *link* que ao ser acionado causa a transição da primeira para a segunda interface como mostra a seta 1 da Figura 6.6a.

A segunda interface simula a monitoração do Domínio identificado pelo nome que está alinhado com o Binóculo. Essa simulação consiste na execução de seis coletas desse Domínio em intervalos de três segundos, finalizando com a apresentação dos colaboradores do Domínio(Agentes SNMP e Estados Primitivos) e das horas dos *pollings* juntamente com os resultados obtidos.

Figura 6.6a - Verificação do funcionamento❶ e da definição❷ de um Domínio

O nome de Domínio, assim como o Binóculo, é um *link* cujo acionamento causa a transição da primeira interface para a terceira interface como mostra a seta 2 da Figura 6.6a. A terceira interface apresenta as especificações que definem o Domínio em questão, isto é, os nomes, tipos e formas dos objetos que representam os Estados Primitivos que compõem um Domínio. Além desses dados, uma mensagem é exibida quando o registro do controle da coleta desse domínio não está concluído. A mensagem é: "O registro desse domínio ainda não foi concluído". Os nomes apresentados são os nomes dos atributos do Repositório do Domínio. Eles são sintaticamente formados pela concatenação de um nome de equipamento com o nome de um Estado Primitivo (apelido de um objeto SNMP) que esse equipamento, enquanto Agente SNMP, pode prover. Na transição indicada pela seta 2 da Figura 6.6a, os objetos *ufcnt_ifInOct_2* e *ufcnt_ipInRec*, cujos tipos e formas são, respectivamente, C(counter), C(colunar) e C(counter), E(escalar), são apresentados.

Finalmente, como último *link* de cada uma das linhas da relação de Domínios apresentada pela primeira interface estão as Ações Inicializar e Finalizar Domínio. A Ação Inicializar causa a execução do processo de monitoração de um Domínio que se encontra na Situação "Inabilitado"(recém definido) ou na Situação "Finalizado". Inversamente, a Ação Finalizar suspende a monitoração de um Domínio que se encontra na Situação "Ativo".

A Figura 6.6b exibe a navegação entre a primeira e a quarta interface através da seta 1. Ela exemplifica a Inicialização do Domínio "ufcntFluxIf2", mostrando tanto o caso de inicialização bem sucedida como o caso de falha na inicialização. As transições para cada um desses casos são indicados pelas setas 2 e 3 respectivamente.

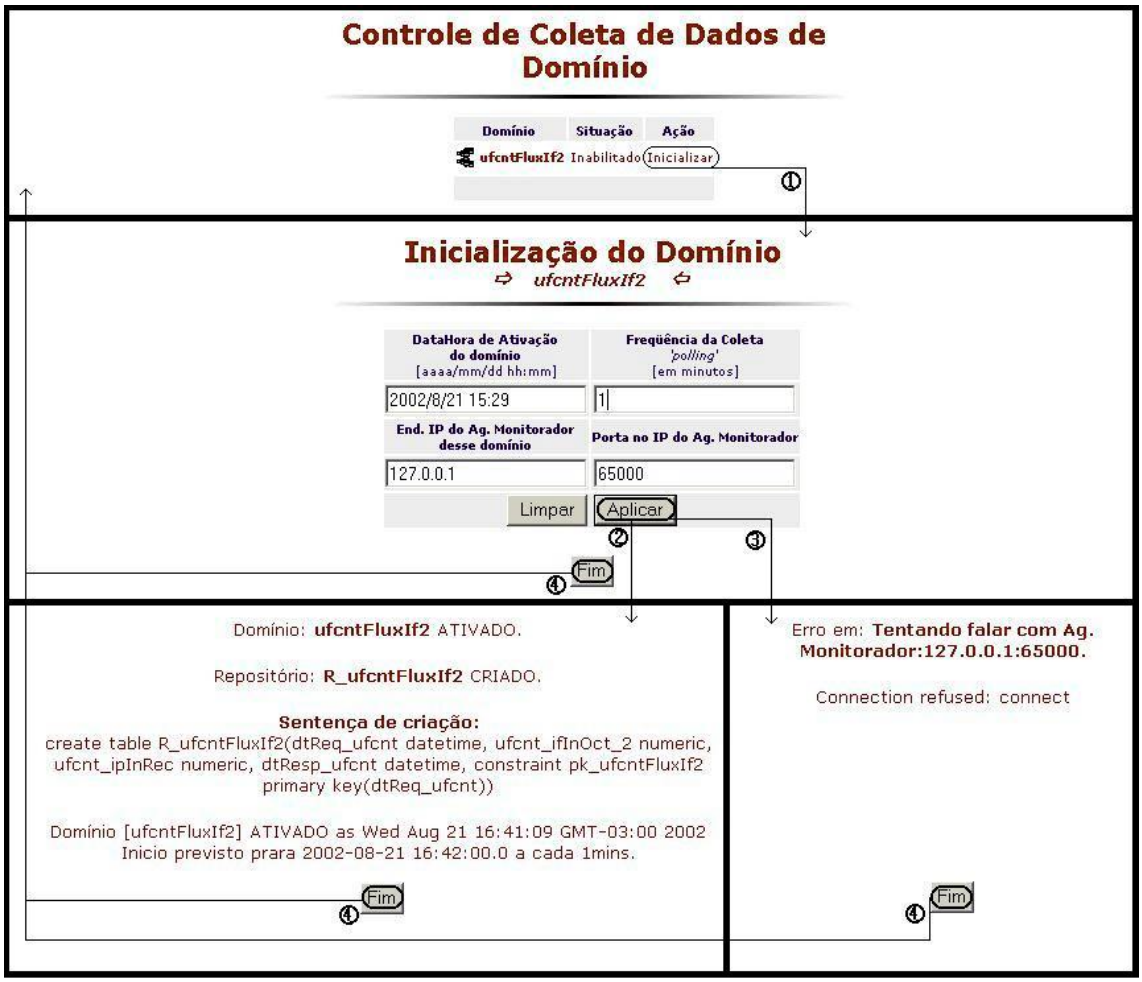

Figura 6.6b - Sucesso❶ e Falha❷ na inicialização de um Domínio

O acionamento da Ação "Inicializar" causa a transição para a quarta interface que registra na tabela Domínio os dados que parametrizam o regime de monitoração de um determinado domínio. Isso ocorre quando botão Aplicar é acionado. Nesse caso, a interface tanto cria a tabela repositório como envia para um certo Agente Monitorador uma mensagem de requisição de Ativação do Domínio associado à Ação escolhida na primeira interface. Nesse momento, o atributo *sitDom* da tabela Domínio recebe o valor "A".

A seta 2 mostra a transição para a interface de apresentação dos resultados da Ação de inicialização ou ativação bem sucedida, executada pela quarta interface. Nela aparecem o nome do domínio inicializado("ufcntFluxIf2"), o nome

da tabela repositório("R_ufcntFluxIf2") do Domínio, a sentença SQL de criação dessa tabela, a data exata de início e a freqüência da monitoração do Domínio.

A seta 3 mostra a transição para a interface de apresentação de falha da inicialização de um Domínio. Isso ocorre quando o Agente Monitorador não está em funcionamento. Nesse caso, são apresentadas as informações(endereço IP e porta TCP) que permitem "identificar" o Agente que não está respondendo. De qualquer modo, os parâmetros que configuram a monitoração ficam registrados na tabela Domínio de forma que o Agente Monitorador pode retornar às coletas quando o mesmo for colocado em execução.

A Ação "Finalizar" é um *link* que causa a transição da primeira para a quinta interface que executa a finalização da monitoração de um Domínio, como mostra a Figura 6.6c. A seta 1 mostra a transição para a interface de finalização bem sucedida, isto é, a quinta interface.

Figura 6.6c - Sucesso❶ e Falha❷ na finalização de um Domínio

O acionamento do *link* "Finalizar" causa a mudança do valor do atributo *sitDom* da tabela Domínio para o valor "F" e envia uma mensagem de requisição de suspensão da monitoração de um certo Domínio ao Agente Monitorador encarregado da sua coleta. A seta 2 mostra a transição para a interface de finalização mal sucedida, isto é, não foi possível haver comunicação com o Agente

Monitorador. No caso de finalização bem sucedida, a interface apresenta o nome do Domínio finalizado, a mensagem enviada, a "identificação" do Agente requisitado e sua resposta. No outro caso, apenas a "identificação" do Agente com o qual o diálogo falhou e o tipo da falha de comunicação ocorrida são apresentados. Isso também porque o Agente não está em execução.

O acionamento do botão Fim, seta 3, sempre faz com que essa transação volte para a primeira interface.

6.3.1.6 Registro de Eventos

A Transação Registro de Eventos provê as funções de manutenção da tabela que implementa a Entidade Fórmula. O seu objetivo é permitir a definição de Eventos Compostos, ou simplesmente Eventos, a partir dos Estados Primitivos que compõem um determinado Domínio. A definição consiste na elaboração de uma fórmula que é utilizada na especificação de uma Visão(*View*) do banco de dados sobre o repositório do Domínio monitorado. A Fórmula define um Evento e a Visão implementa o mecanismo para sua percepção. Essa transação apresenta três interfaces.

A função da primeira interface é apresentar todos os Eventos registrados, abrindo a possibilidade para a criação de um novo Evento e a possibilidade da simples visualização de um Evento já definido. A segunda interface tem como função fazer a definição de um Evento. Ela auxilia na elaboração da Fórmula que o expressa ao mesmo tempo em que cria, dinamicamente, a especificação da Visão que provê o mecanismo de percepção desse Evento. E a função da terceira interface é apresentar a definição de um certo Evento, dando a opção de remoção da sua Visão e do seu registro da tabela Fórmula simultaneamente.

A Figura 6.7a exibe através das setas a navegação típica entre a interfaces primeira e segunda dessa transação. Ela exemplifica o registro do Evento chamado "ufcntFluxSuspIf2".

Na primeira interface, é possível especificar um novo Evento a ser criado juntamente com o Domínio onde ele possivelmente ocorra ou apenas selecionar um Evento, dentre aqueles já registrados, para a visualização da sua definição. Em caso de criação, a transição para a segunda interface é causada pelo acionamento do botão Avançar como mostra a seta 1.

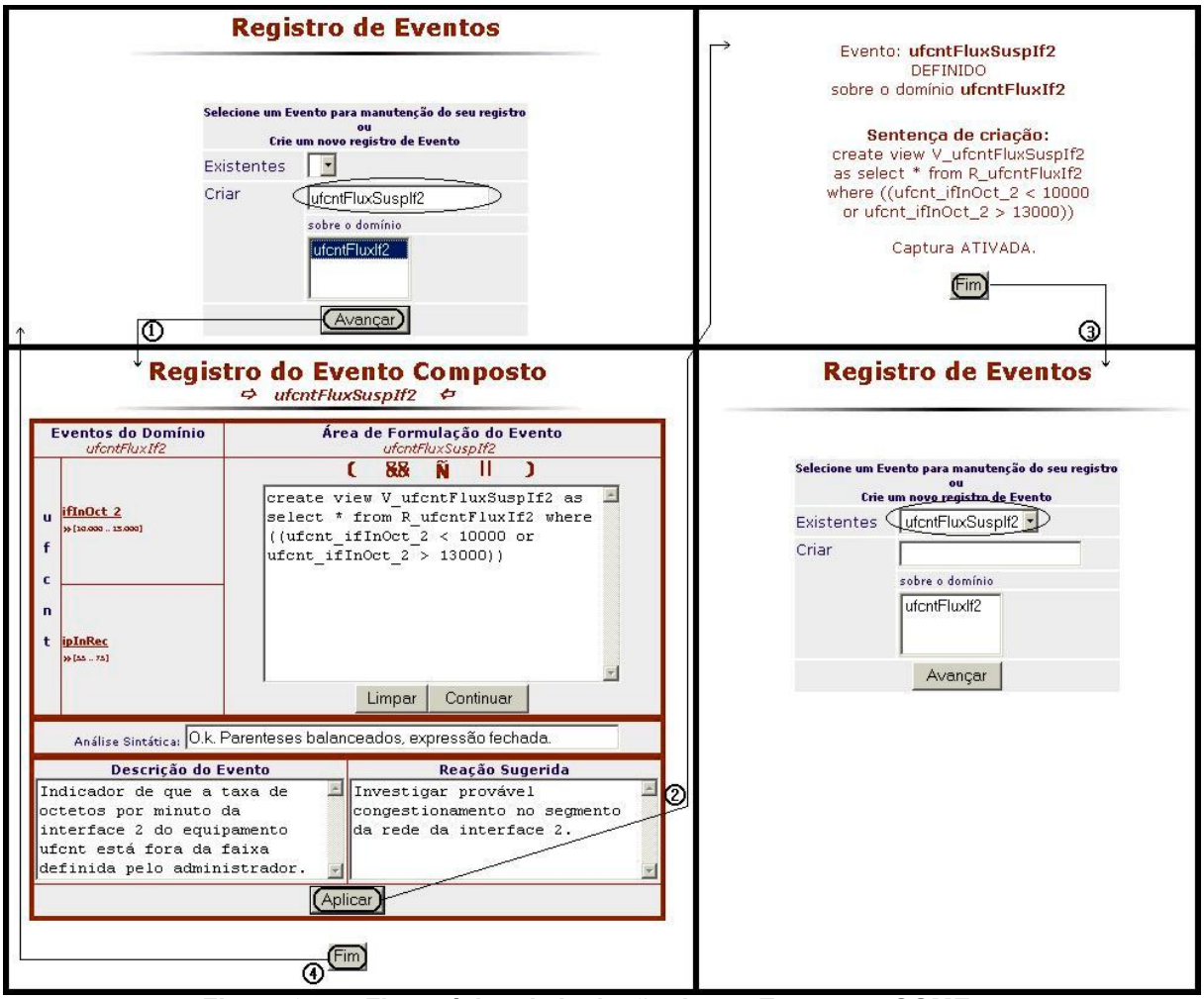

Figura 6.7a - Fluxo típico de inclusão de um Evento no SGME

A segunda interface apresenta um formulário com toda a composição do Domínio escolhido na primeira interface. Ela auxilia a formulação sintática do Evento e, conseqüentemente, da Visão por meio de simples 'cliques' sobre os ícones Parênteses[**( )**], E[**&&**], Não[**Ñ**], Ou[**||**], e sobre os nomes dos Estados

Primitivos cujas Restrições já estão implícitas(Registro de Estado Primitivo). Esse processo de formulação é controlado de forma que sempre resulta uma expressão lógica ou fórmula bem formada, que é utilizada na expressão da sentença SQL de criação da Visão que permite ao *framework* prover o meio para a percepção do Evento. Mensagens indicativas dos eventuais erros sintáticos cometidos nesse processo são apresentadas dinamicamente. Além disso, informações descritivas do Evento e uma sugestão documentacional de reação para controle das suas causas também podem ser registradas. O acionamento do botão Aplicar só é possível se a definição do Evento(sintaxe da Fórmula) estiver correta. Nesse caso, uma interface de simples ratificação do registro e criação da Visão do Evento é apresentada imediatamente como mostra a seta 2 da Figura 6.7a.

No caso de visualização, a terceira interface apenas apresenta o nome do Evento, o nome do Domínio monitorado, as descrições do Evento e da reação sugerida, e a sentença SQL de criação da Visão na qual está embutida a Fórmula que define o Evento. A sentença SQL tem a seguinte estrutura sintática:

"create view " +
"V_" + <nome_do_evento> + " as select * from " +
"R_" + <nome_do_domínio> + " where " + <fórmula>.

A transição da primeira interface para a terceira é causada pelo acionamento do botão Avançar após a escolha de um Domínio como mostra a seta 1 da Figura 6.7b.

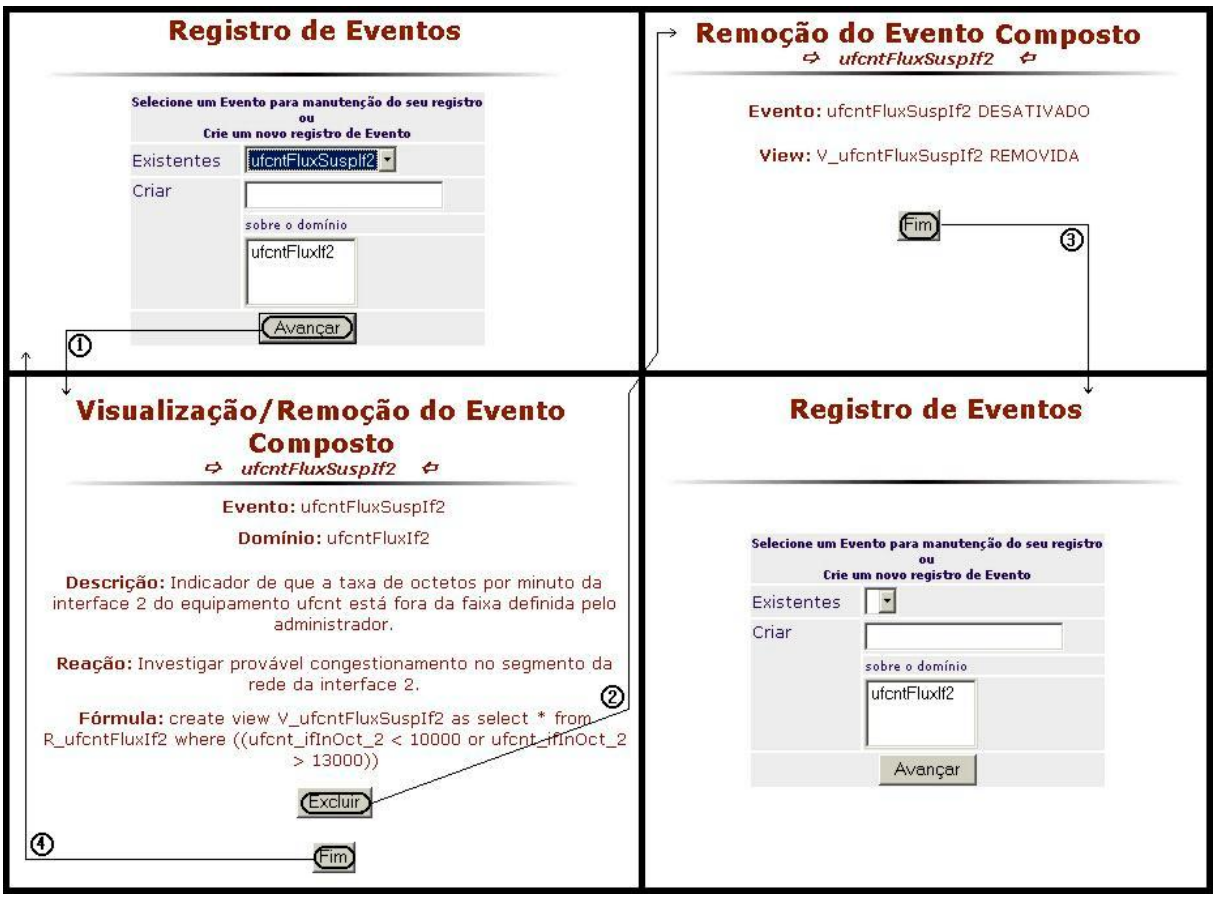

Figura 6.7b - Fluxo típico de exclusão de um Evento do SGME

Opcionalmente, é possível acionar o botão Excluir para remover tanto o registro do Evento exibido como a definição da sua Visão. A seta 2 da Figura 6.7b mostra a transição para a interface que apresenta a mensagem que ratifica a remoção do Evento.

O acionamento do botão Fim, em qualquer momento dessa transação, ocasiona a apresentação da primeira interface. Quando um Evento é excluído, ele deixa de aparecer na relação dos já existentes. Essas transições podem ser acompanhadas por meio das setas 3 e 4 da Figura 6.7b.

6.3.2 A Definição e Geração de Relatórios Gerenciais

A Definição e Geração de Relatórios Gerenciais são feitas pelas transações Visor de Repositório e Visor de Evento.

6.3.2.1 O Visor de Repositório

A Transação Visor de Repositório provê a função de consulta aos repositórios dos Domínios. O seu objetivo é produzir relatórios a partir dos Estados Compostos coletados. Essa transação apresenta três interfaces.

A função da primeira interface é apresentar todos os Domínios ativos, abrindo a possibilidade para a elaboração de relatórios a partir dos dados armazenados nos seus repositórios. A segunda interface tem como função auxiliar a construção de sentenças SQL de consultas aos repositórios. E a função da terceira interface é executar e apresentar os resultados dessas consultas.

A Figura 6.8 exibe através das setas a navegação entre as interfaces dessa transação. Ela exemplifica uma consulta ao repositório do Domínio "ufcntFluxIf2".

A primeira interface apresenta todos domínios ativos. Aqueles que estão sendo coletados por algum Agente Monitorador. A escolha de um Domínio seguida do acionamento do botão Avançar provoca a transição da primeira para a segunda interface como indica a seta 1 da Figura 6.8.

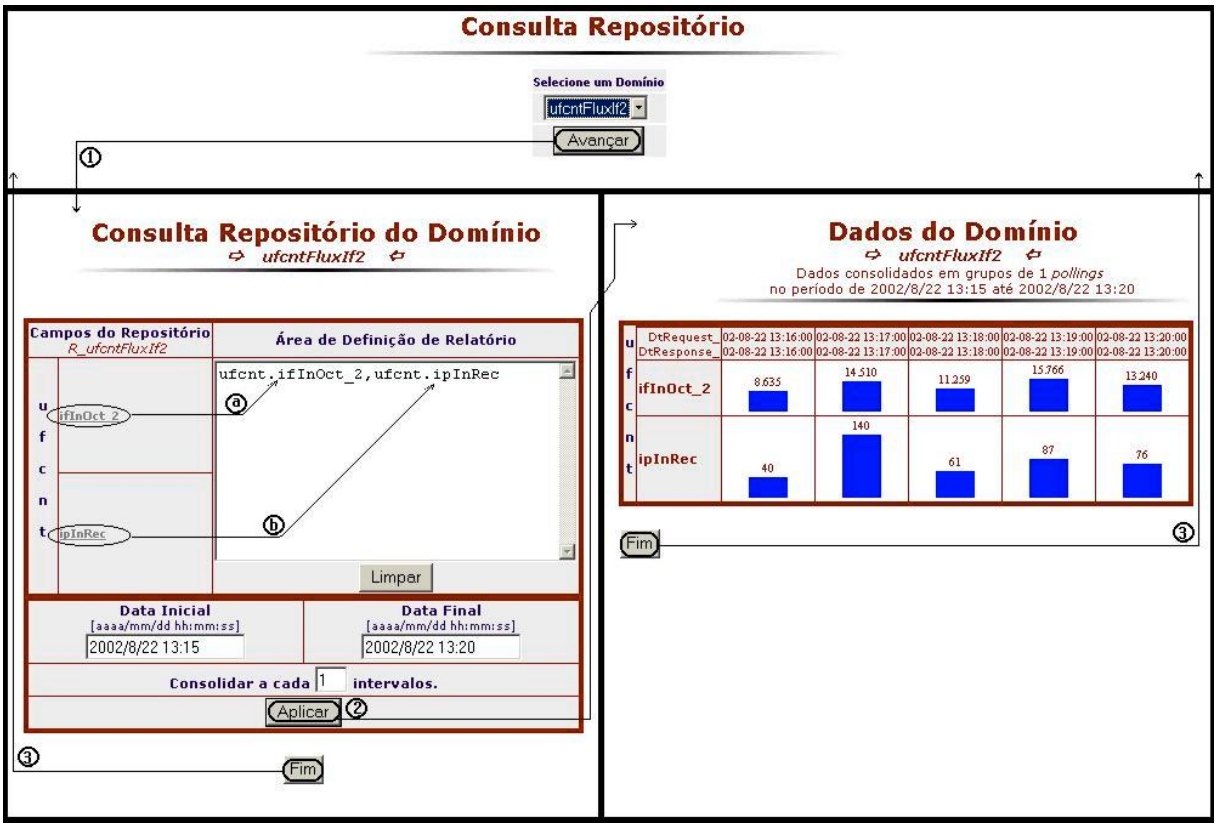

Figura 6.8 - Fluxo típico de definição de um relatório no SGME

A segunda interface apresenta um formulário com toda a composição do Domínio escolhido na primeira interface. Ela auxilia a construção da expressão sintática de uma sentença SQL de consulta ao repositório("R_ufcntFluxIf2") do Domínio em questão, permitindo que os Estados Primitivos que vão compor o relatório desejado sejam determinados por meio de simples 'cliques' sobre seus nomes como mostram as setas a e b. Também é possível filtrar, cronologicamente, o período de tempo cujos dados serão retratados no relatório e estabelecer o modo de consolidação desses dados. A consolidação consiste em exibir os valores dos Estados Primitivos cumulativamente a cada *n* registros lidos do repositório. Nesse caso, cada valor exibido representa o somatório de cada *n* valores cronologicamente contíguos armazenados no banco de dados dentro do período de tempo especificado.

O acionamento do botão Aplicar causa a transição para a terceira interface que executa a sentença SQL definida, consolida os dados conforme especificado e apresenta o relatório conforme indica a seta 2 da Figura 6.8. Essa interface apresenta os resultados da consulta tanto na forma textual como na forma de gráficos de barras, claro, apenas, quando se tratar de conteúdos de tipos numéricos.

6.3.2.2 O Visor de Evento

A Transação Visor de Eventos provê a função de consulta às Visões dos Eventos definidos dentro do *framework*. Ela ilustra a utilização do mecanismo de percepção de eventos. O seu objetivo é apenas relatar as ocorrências dos eventos ocorridos sobre um determinado Domínio. Essa transação apresenta duas interfaces.

A função da primeira interface é apresentar todos os Domínios ativos, abrindo a possibilidade para a elaboração de relatórios sobre os Eventos definidos sobre um certo Domínio. A segunda interface tem como função executar consultas às Visões que possibilitam perceber a ocorrência ou não de Eventos.

A Figura 6.9 exibe, através das setas, a navegação entre as interfaces dessa transação. Ela exemplifica a observação das ocorrências de todos os Eventos definidos sobre o Domínio "ufcntFluxIf2". Nesse caso, apenas um Evento, "ufcntFluxSuspIf2", está sendo observado pelo SGME.

Na primeira interface, são apresentados todos os nomes dos Domínios ativos, do mesmo modo que na transação Visor de Repositórios. Essa interface permite a escolha de um Domínio e a especificação de um filtro temporal para que a transação relate todos os Eventos que o SGME percebeu dentro de um certo intervalo de tempo. O acionamento do botão Avançar causa a transição para a segunda interface como mostra a seta 1.

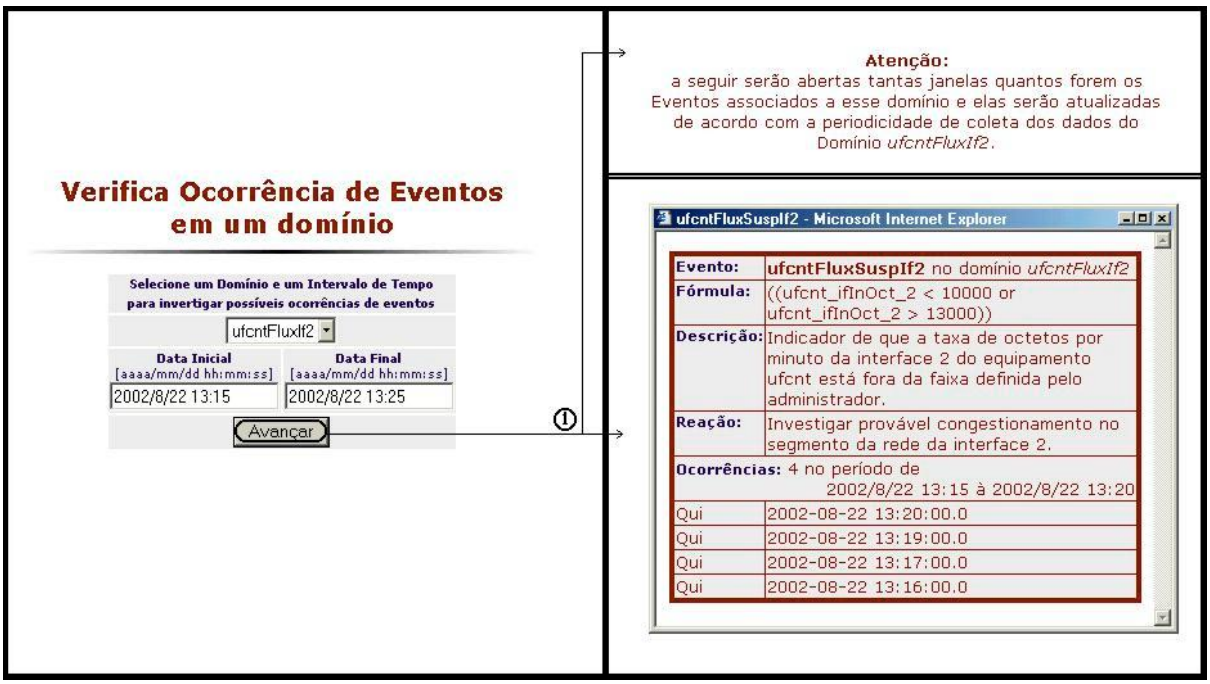

Figura 6.9 - Exemplo de verificação de ocorrência de um Evento no SGME

De acordo com as especificações do MDGE, para cada Domínio, podem estar definidos vários Eventos cujas ocorrências podem ser capturadas por meio das suas respectivas Visões. Desse modo, são apresentadas tantas versões da segunda interface quanto seja o número de Eventos definidos para o Domínio escolhido na primeira interface. Nessas versões, são exibidos o nome do Evento, o Domínio afetado, a Fórmula e a descrição do Evento, a reação sugerida, o período de tempo analisado, a quantidade juntamente com as precisas datas de ocorrências dos Eventos. Cada uma dessas versões são atualizadas e reapresentadas de acordo com a freqüência especificada para o *polling* do Domínio monitorado.

Além disso, a segunda interface apresenta, inicialmente, uma mensagem fixa, expressando um resumo do parágrafo anterior, enquanto que as versões da segunda interface são apresentadas superpostas sobre a interface primária do *framework*.

6.3.3 O Processo de Monitoração

O mapeamento das informações das MIBs dos diversos agentes SNMP em um banco de dados relacional juntamente com as funcionalidades providas pelas transações de operacionalização do SGME ainda não dão plenitude ao funcionamento do *framework* como está proposto. Para que possa atingir os seus objetivos monoliticamente, falta um componente que vai dar transparência ao processo de monitoração aos possíveis sistemas de gerenciamento de redes de computadores que, porventura, venha a suportar.

Esse componente é o Agente Monitorador. Ele executa o processo de monitoração da rede que envolve a identificação, a coleta e o armazenamento de grupos de informações gerenciais, Domínios, a partir dos dados registrados no banco de dados do SGME. Com isso, o processo de construção de um sistema de gerenciamento pode concentrar-se apenas no acesso aos Repositórios e às Visões dos Domínios monitorados para proceder as ações administrativas necessárias, dentro do contexto do gerenciamento de redes baseado no protocolo SNMP. O seu objetivo é determinar Quais, Onde, Quando e Como os objetos SNMP codificados como Estados Primitivos devem ser monitorados.

A execução do Agente Monitorador ocorre em dois momentos funcionais distintos. Primeiro, ele identifica e retoma a monitoração de todos os Domínios Inicializados ou Ativos, a seguir, ele permanece constantemente de prontidão para atender solicitações do tipo 'Inicializar' e/ou 'Finalizar' Domínio, oriundas da Transação Controle da Coleta, até que receba uma mensagem do tipo 'Fim' para encerrar a sua execução.

No primeiro momento, são identificados todos os Domínios que devem ser monitorados por ele, isto é, todos aqueles registrados na tabela Domínio cujos atributos *sitDom* e *endAgMonDom* estão valorados conforme a tabela 6.1. Com isso, o Agente Monitorador pode ser distribuído e "localizado" pelo SGME em qualquer local das diversas sub-redes gerenciadas.

Tabela 6.1 - Parâmetros de inicialização dos *pollings* de um Domínio

Atributo	Valor
sitDom	'A'
endAgMonDom	endereço IP do *host* do Agente Monitorador

Uma vez de posse desses registros, o Agente Monitorador constrói um objeto que fica encarregado de executar as monitorações de cada um dos Domínios selecionados. Cada objeto desses funciona em duas fases.

Na primeira fase, ele recupera das tabelas Domínio, Comunidade, EstadoComposto, EstadoPrimitivo e Índice as informações relativas aos Domínios ativos que permitem:

- Identificar o repositório do Domínio que vai monitorar. O nome do repositório é inferido por meio da concatenação de 'R_' com o nome do Domínio;
- Inferir os nomes e os tipos de dados dos atributos do repositório;
- Montar a sentença do tipo *Prepared SQL* de inserção de dados no repositório. Nesse momento, apenas a parte da sintaxe dessa sentença que especifica o nome da tabela repositório e os nome dos seus atributos;
- Montar os OIDs dos objetos SNMP codificados como Estados Primitivos que compõem o Domínio;
- Estabelecer as conexões com os Agentes SNMP provedores do Domínio.

Na segunda fase, o objeto de monitoração dispara uma *thread* que executa o processo de monitoração propriamente dito, isto é, inicia a coleta e o armazenamento do Domínio a partir da data e com a freqüência registrados no banco de dados até que uma solicitação do tipo 'Finalizar' Domínio seja recebida pelo Agente Monitorador. No caso da impossibilidade de obtenção de parte ou de todas as informações de um Domínio, então os atributos numéricos serão valorados com -1 e os demais com *null*.

No segundo momento da sua execução, o Agente Monitorador disponibiliza um *socket,* quer dizer, uma porta TCP para trocar mensagens com a Transação Controle da Coleta e também com aplicativos do tipo *telnet,* nesse caso, porém, apenas para fins de depuração da sua implementação.

O Agente Monitorador é capaz de receber as seguintes mensagens:

- 'ative' para iniciar a coleta de um certo Domínio por meio da construção de um objeto de monitoração para ele. O formato dessa mensagem é: 'ative <nome_do_domínio>';
- 'aborte' para finalizar imediatamente a coleta de um certo Domínio por meio do cancelamento da *thread* que está encarregada pela monitoração dele. O formato dessa mensagem é: 'aborte <nome_do_domínio>';
- 'liste' para solicitar a relação dos Domínios que estão sendo monitorados no momento da recepção dessa mensagem. Tem fins depurativos e o seu formato é: 'liste';
- 'datahora' para requisitar a data e a hora, com precisão de milisegundos, correntes no *host* do Agente Monitorador. O formato dessa mensagem é 'datahora' e o seu retorno tem o seguinte formato: '**#ano/mes/dia hora:minuto:segundo#milésimo_de_segundo**'. O seu propósito é facilitar o sincronismo entre uma aplicação de gerenciamento que se deseje construir e o processo de coleta de dados executado pelo Agente Monitorador;
- 'fim' para parar a execução do Agente Monitorador;
- '?' ou 'b' para solicitar ajuda sobre essas mensagens e suas funções. Tem fins apenas documentacionais.

Nesse segundo momento, então, o Agente Monitorador fica em constante estado de prontidão, "escutando" a sua Porta TCP, aguardando pela recepção de algum desses tipos de solicitações ao mesmo tempo em que mantém as suas *threads* executando as sua devidas monitorações.

O Agente Monitorador é codificado na linguagem Java, o que o torna capaz de ser executado em todas as plataformas de sistemas operacionais modernos disponíveis no mercado. Além disso, o MDGE está projetado de forma que dá condições ao Agente Monitorador de ter várias cópias sendo executadas simultaneamente em diversas sub-redes de computadores que se deseja gerenciar.

Com esse componente, o *framework* SGME passa a ter, finalmente, condições de alcançar todos os objetivos a que se propõe.

Capítulo 7 - Conclusão

- O *Framework* SGME

Esse trabalho, apresenta a implementação do *framework* SGME, como ferramenta de apoio ao desenvolvimento de aplicativos dirigidos a gestão dos vários Domínios de Gerenciamento que podem surgir no cenário das redes de computadores, monitoráveis com o auxílio do protocolo SNMP.

O SGME automatiza o agrupamento de informações de gerenciamento dos Agentes SNMP, produzindo estruturas de dados que representam Domínios de Gerenciamento e expressões ou regras, formuladas com essas informações, que representam Eventos. Essas estruturas são transformadas em repositórios de dados e as Fórmulas em mecanismos de percepção de Eventos. Os Domínios de Gerenciamento são reconhecidos por um Agente Monitorador que passa a coletar as informações que os compõem, armazenando-os em um banco de dados.

- As Tecnologias

A implementação do SGME está baseada nas tecnologias de Modelagem de Domínios, Modelagem de Eventos, Modelagem de Banco de Dados, Modelagem Web e Modelagem Cliente/Servidor.

A Modelagem de Domínios fundamenta a delimitação das fronteiras físicas e administrativas das ações de gerenciamento sobre uma rede. Essa delimitação é feita por meio da definição de estruturas de dados que culmina com a criação automática de tabelas em um banco de dados que representam essas estruturas.

A Modelagem de Eventos é feita por meio de regras que balizam os atributos das tabelas onde os Domínios são armazenados. Essas expressões permitem a criação de Visões do banco de dados que podem ser utilizadas como mecanismos de percepção de Eventos.

A Modelagem de Banco de Dados permite que os Domínios sejam representados como tabelas em um Sistema Gerenciador de Banco de Dados Relacional. Ele proporciona a utilização de estruturas de armazenamento isentas das limitações de espaço que sofrem as sondas RMON com o uso de *buffers* circulares. Com isso, as aplicações de gerenciamento podem ter acesso, de modo mais amigável, às informações armazenadas pelo Agente Monitorador no banco de dados, por meio de sentenças SQL. Dessa forma, elas podem se abstrair dos *pollings* de obtenção de informações dos Domínios que devem gerenciar e se dedicar, exclusivamente, a processar dados com vistas a tomada de alguma ação de gerenciamento. A tecnologia de SGBDR também provê o recurso de Views(Visões), utilizado para percepção de Eventos, além de *Triggers* e *Stored Procedures* cuja utilização poderia ser investigada para a expansão dos serviços de apoio às aplicações de gerência prestados pelo SGME.

As Modelagens Web e Cliente/Servidor fundamentam a capacidade de distribuição dos serviços prestados pelo *framework*. O projeto das interfaces baseado na Web facilita a utilização do SGME pelo fato de incorporarem objetos que já são comumente reconhecidos pelos usuários da Internet. Os serviços providos tornam-se acessíveis a partir de qualquer ponto da rede.

- Abstração de Plataformas

A conjugação dessas modelagens é feita pela implementação dos serviços prestados pelo *framework*. Esses serviços são implementados por meio de pequenos programas codificados em Java: as Transações e o Agente Monitorador. Mais especificamente, as Transações são implementadas como Java *servlets*. A utilização do SGBDR, do Servidor WEB e da Biblioteca AdventNet selecionados, juntamente com a linguagem Java, confere ao SGME um alto grau de portabilidade.

- Os Objetivos do SGME

Os serviços SGME provêem a comunicação entre um Administrador e os nodos gerenciados da rede, a definição de Domínios de Gerenciamento, a monitoração dos nodos da rede, a construção dinâmica e transparente de repositórios para os dados monitorados, a definição de eventos e a produção dinâmica e transparente de relatórios gerenciais.

- Complexidade e Custo

Dada a simplicidade da sua arquitetura, com uma quantidade mínima de componentes, o SGME se constitui numa solução de baixo custo já que a maior parte dos seus componentes são muito provavelmente legados, no que diz respeito às suas características funcionais.

- Potencialidades do SGME

O Agente Monitorador faz os *pollings* dos Domínios por meio de *threads* distintas. Isso, aliado a uma definição bem objetiva de cada Domínio e uma boa freqüência de coleta para cada um deles, pode proporcionar um certo grau de otimização da coleta de dados dos domínios monitorados.

O SGME apresenta um potencial como ferramenta de treinamento no protocolo SNMP. A configuração dos processos de monitoração requer que o usuário passe pela definição de comunidades, manipule, alimentando e indexando, OIDs, crie Domínios agrupando esses OIDs, indo até a ativação da coleta dos Domínios. Isso se constitui numa prática que perpassa uma boa parte dos elementos básicos do protocolo SNMP.

- Trabalhos Futuros

Como trabalhos futuros poder-se-ia verificar a possibilidade de implementação da idéia de coleta baseada em Domínios e do conceito de Eventos baseados em visões (*views*) de bancos de dados diretamente no protocolo SNMP

dos Agentes e das sondas RMON. A utilização de *Triggers* e *Stored Procedures*, na automatização de procedimentos reativos, também poderiam ser investigados.

No caso das redes convencionais, assim como hoje configuram-se por exemplo portas e comunidades diretamente no nodo instrumentalizado com o protocolo SNMP, também poder-se-ia especificar o banco de dados, onde estariam contidas as definições dos Domínios que se deseja monitorar. Dessa maneira, o protocolo obteria localmente os dados dos domínios, armazenando-os diretamente no banco de dados especificado.

Já, no caso das redes ativas, o Agente Monitorador passaria a lançar e a sincronizar a execução das suas *threads* para os Agentes SNMP e/ou sondas RMON. Neste contexto, as *threads* constituem pacotes ativos de coleta de dados dos Domínios monitorados. Novamente, teríamos a obtenção local dos dados com armazenamento direto em um banco de dados.

Em ambos os casos, têm-se como expectativa, além da continuidade do apoio a construção de aplicações de gerenciamento alimentadas por bancos de dados relacionais, a diminuição do tráfego de mensagens SNMP na rede e a não utilização do protocolo UDP.

Referências Bibliográficas

[ABOELELA 1999] ABOELELA, Emad, DOULIGERIS, Christos. “Fuzzy Temporal Reasoning Model for Event Correlation in Network Management”. *Conference on Local Computer Networks LCN '99* , p150 -159, Out 1999.

[ADVENTNET 2002] ADVENTNET. *“Adventnet Products Overview”*. Produced by AdventNet Inc. Disponível em *<http://www.adventnet.com/products/index.html>.* Acesso em: 20/12/2002 .

[AHN 1999] AHN ,Seong Jin, YOO,Seung Keun, CHUNG,Jin Wook. ”Design and Implementation of a Web-Based Internet Performance Management System using SNMP MIB-II”.*International Journal of Network Management*; v.9, n.5, p309-321, Set-Out 1999.

[BASHIR 1998] BASHIR, Omar, PHILLIPS, Iain, PARISH, David. “Warehousing Communication Network Monitoring Data”. *IT Strategies for Information Overload (Digest No: 1997/340)*, p10/1 -10/4, Dez 1998.

[BAYARDO 1997] BAYARDO, R. J., BOHRER, W. Jr., BRICE, R. [et al]. “InfoSleuth: Agent-Based Semantic Integration of Information in Open and Dynamic Environments”. *Proceedings of the 1997 ACM SIGMOD International Conference on Management of Data*, Arizona, Jun 1997.

[BOHORIS 2000] BOHORIS, C., PAVLOU, G., CRUICKSHANK, H. “Using Mobile Agents for Network Performance Management”. In *IEEE/IFIP Network Operations and Management Seminar NOMS 2000,* Honolulu, Mai,2000.

[BOUDAOUD 2000] BOUDAOUD, K., LABIOD, H., BOUTABA, R. [et al].”Network Security Management with Intelligent Agents”. In *IEEE/IFIP Network Operations and Management Seminar NOMS 2000*, Honolulu, Mai,2000.

[BURGESS 2000] BURGESS, John, GUILLERMO, Ray. “Raising Network Fault Management Intelligence”. In *IEEE/IFIP Network Operations and Management Seminar NOMS 2000,* Honolulu, Mai,2000.

[CASE 2001] CASE, Jeff. "SNMP Update" SNMP Research. North American Network Operators' Group - NANOG 22, May 2001.

[CHEIKHROUHOU 2000] CHEIKHROUHOU, M., LABETOULLE, J. ”An Efficient Polling Layer for SNMP”.In *IEEE/IFIP Network Operations and Management Seminar NOMS 2000,* Honolulu, Mai,2000.

[COOK 1998] COOK, Jonathan E., WOLF, Alexander L. “Discovering Model of Software Process from Event-Based Data”. *ACM Transactions on Software Engineering and Methodology*, v. 7, n.3, p215-249, Jul 1998.

[CORTE 2001] CORTE, Aurelio La, PULIAFITO,Antonio, TOMARCHIO,Orazio. ”QoS management in programmable networks through mobile agents”. *Microprocessors and Microsystems* v.25, n.2, p111-120, Abr 2001.

[DER 1999] DERI, Luca.“Desktop-based Network Management”. In *Sixth IFIP/IEEE International Symposium on Integrated Network Management IM’99,* Boston, Mai,1999.

[DERI 1999] DERI,Luca. “Desktop versus Web-based Network Management”. *International Journal of Network Management*; v.9, n.6, p371-378, Nov-Dez 1999.

[DUARTE 2001] DUARTE JR., Elias Procópio, SANTOS, Aldri L. dos. “Network Fault Management Based on SNMP Agent Groups”. *International Conference on Distributed Computing Systems Workshop 2001*, p51-56, Abr 2001.

[FESTER 1999] FESTER , O., FESTOR, P., YOUSSEF, N.Ben [et al]. “Integration of WBEM-based Management Agents in the 0SI Framework”. In *Sixth IFIP/IEEE International Symposium on Integrated Network Management IM’99,* Boston, Mai,1999.

[GAVALAS 2000] GAVALAS, Damianos, GHANBARI, Mohammed., O'MAHONY, Mike [et al]. “Enabling Mobile Agent Technology for Intelligent Bulk Management Data Filtering”. In *IEEE/IFIP Network Operations and Management Seminar NOMS 2000*, Honolulu, Mai,2000.

[GUPTA 1999] GUPTA,Alok, STAHL,Dali O., WHINSTON, Andrew B. “The economics of network management”. *Communications of the ACM*, v 42, n 9, p57-63, Set 1999.

[GHETIE 1997] GHETIE, Iosif G. *Networks and systems management: platforms analysis and evaluation.* New Jersey: Kluwer Academic, c1997, 505p.

[HARIRI 2000] HARIRI, Salim, KIM, Yoonhee. "Design and Analysis of a Proactive Application Management System (PAMS)". In *IEEE/IFIP Network Operations and Management Seminar NOMS 2000*, Honolulu, Mai,2000.

[HASAN 1999] HASAN, Masum, SUGLA, Binay e VISWANATHAN, Ramesh."A Conceptual Framework for Network Management Event Correlation and Filtering Systems". In *Sixth IFIP/IEEE International Symposium on Integrated Network Management IM'99,* Boston, Mai,1999.

[H 1999] HO, L.Lawrence, CAVUTO, David.J, HASAN, M.Z [et al]. "Adaptive Network/Service Fault Detection in Transaction-Oriented Wide Area Networks". In *Sixth IFIP/IEEE International Symposium on Integrated Network Management IM'99*, Boston, Mai,1999.

[HO 1999] HO L.Lawrence, CAVUTO, David.J, PAPAVASSILIOU, Symeon [et al]. "Adaptive and Automated Detection of Network/Service Anomalies in Wide Area Networks". In *Journal of Network and Systems Management,* p761-765, 1999.

[HO 2000] HO L.Lawrence, CAVUTO, David.J, PAPAVASSILIOU, Symeon [et al]. "Adaptive and Automated Detection of Network/Service Anomalies in Transaction-Oriented WAN´s:Network Analysis, Algorithms Implementation and Deployment". In *IEEE Networks Journal;* v. 18, n. 5, p256-268, Mai 2000.

[JAKOBSON 2000] JAKOBSON, G., WEISSMAN, M., BRENNER, L. [et al]. "GRACE: Building Next Generation Event Correlation Services". In *IEEE/IFIP Network Operations and Management Seminar NOMS 2000*, Honolulu, Mai,2000.

[JONES 1999] JONES, G., ZEISLER, E., CHEN, L. "Web-based Messaging Management Using Java Servlets". In *Sixth IFIP/IEEE International Symposium on Integrated Network Management IM'99,* Boston, Mai,1999.

[JONES 1997] JONES, Katherine. "ServerWORKS Manager: Network Management Integrated with Enterprise and Applications Servers". *International Journal of Network Management*; v.7, n.6, p334-338, Nov-Dez 1997.

[JU 2000] JU, Hong-Taek, CHOI, Mi-Joung, HONG, James W. ”An efficient and Lightweight embedded Web server for Web-based Network Element Management”. *International Journal of Network Management*; v.10, n.5, p261-275, Set-Out 2000.

[KATZELA 1995] KATZELA, Irene, SCHWARTS,Mischa. “Schemes for Fault Identification in Communication Networks”. *IEEE/ACM Transactions on Networking*, v.3, n.6, p753-764, Dez 1995.

[KELLER 1998] KELLER,Rudolf K., TESSIER,Jean, VON BOCHMANN, Gregor. “A Pattern System for Network Management Interfaces”. *Communications of the ACM*, v.41, n. 9, p86-93, Set 1998.

[KIM 2000] KIM,Do-Hyeon, CHO,You-Ze. ”Design and implementation of network management systems for integrated management of LANs and WANs”. *International Journal of Network Management*; v.10, n.3, p135-143, Mai-Jun2000.

[KING 2000] KING,A., HUNT,R. “Protocols and architecture for managing TCP/IP network infrastructures”. *Computer Communications*; v.23, n.16, p1558-1572, Set 2000.

[KLEMETTINEN 1994] KLEMETTINEN,Mika, MANNILA,Heikki, RONKAINEN,Pirjo [et al]. “Finding Interesting Rules from Large Sets of Discovered Association Rules”. *Proceedings of the Third International Conference on Information and Knowledge Management*, Maryland, Nov 1994.

[KNOBBE 1999] KNOBBE, A., WALLEN, D. Van Der e LEWIS, Lundy. ”Experiments with Data Mining in Enterprise Management”. In *Sixth IFIP/IEEE International Symposium on Integrated Network Management IM’99,* Boston, Mai,1999.

[KOHLI 2000] KOHLI,Madhur, LOBO, Jorge. “Distributed actions plans in an agent based network management system”. *Proceedings of the Fourth International Conference on Autonomous Agents*, Barcelona, 2000.

[LEWANDOWSKI 1998] LEWANDOWSKI, Scott M. “Frameworks for components-based clients/server computing”. *ACM Computing Surveys*, v.30, n.1, p3-27, Mar 1998.

[LI 2000] LI, Jung-Shian.”Measurement and in-service monitoring for QoS violations and spare capacity estimations in ATM network”. *Computer Communications*; v.23, n.2, p162-170, Jan 2000.

[LIU 1999] LIU, G., MOK, A. K. e YANG, E. J. ”Composite Events for Network Event Correlation” In *Sixth IFIP/IEEE International Symposium on Integrated Network Management IM’99*, Boston, Mai,1999.

[LO 1998] LO, Chi-Chum, CHEN Shing-Hong. “Robust Event Correlation Scheme for Fault Identification in Communications Network”. *Global Telecommunications Conference GLOBECOM 98,* v.6, p3745-3750,1998.

[MARTIN-FLATIN 1999] MARTIN-FLATIN, J.P.”Push vs. Pull in Web-Based Network Management”. In *Sixth IFIP/IEEE International Symposium on Integrated Network Management IM’99,* Boston, Mai,1999.

[MELCHIORS 2000] MELCHIORS, C., TAROUCO, L.M.R. ”Troubleshooting Network Faults Using Past Experience”. In *IEEE/IFIP Network Operations and Management Seminar NOMS 2000*, Honolulu, Mai,2000.

[MULLER 1997] MULLER, Nathan J. “Web-accessible Network Management Tool”. *International Journal of Network Management*; v.7, n.5, p288-297, Set-Out 1997.

[MYSQL 2002] MYSQL. *“MySql Downloads”*. Produced by MYSQL AB. Disponível em <*http://www.mysql.com/downloads/index.html*>**.** Acesso em: 20/12/2002 .

[NETCRAFT 2002] NETCRAFT. *“Netcraft Web Server Survey”.* Produced by Netcraft. Disponível em
< *http://www.netcraft.com/survey/index-200207.html#active*>**.** Acesso em: 20/12/2002 .

[ORFALI 1996] ORFALI, Robert, HARLEY, Dan, EDWARDS, Jeri. *The essential Client/Server: survival guide.* Canadá: Wiley, c1996,675p.

[PAGUREK 2000] PAGUREK, B., WANG, Y., T. WHITE. “Integration of Mobile Agents with SNMP: Why and How”. In *IEEE/IFIP Network Operations and Management Seminar NOMS 2000*, Honolulu, Mai,2000.

[PARULKAR 1997]PARULKAR, G., SCHMID, D., KRAEMER, C. E. [et al]. “An Architecture for Monitoring, Visualization, and Control of Gigabit Networks”.In *IEEE Networks*, Sept/Out.

[PAPAVASSILIOU 1998] PAPAVASSILIOU, Symeon., SAVANT, V.S., TUPINO, J.J. [et al]. “Enhanced Network Management for Online Services”. In *Proc. IEEE International Conference on Computer Communications and Network*s IC3N’98, Louisian, Oct,1998.

[PENIDO 1999]PENIDO, G., NOGUEIRA, J.M, MACHADO, C. “An automatic fault diagnosis and correction system for telecommunications management”. In *Sixth IFIP/IEEE International Symposium on Integrated Network Management IM’99,* Boston, Mai,1999.

[PERKINS 1997] PERKINS,David, MCGINNIS, Evan. *Understanding SNMP MIBs*.New Jersey: Prentice Hall PTR, c1997. 511p.

[PULIAFITO 2000] PULIAFITO,A., TOMARCHIO,O. “Using mobile agents to implement flexible network management strategies”. *Computer Communications*; v.23, n.8, Abr 2000, p708-719.

[RAZ 2000] RAZ,D., SKGLA,B. “Economically Managing Multiple Private Data Networks”. In *IEEE/IFIP Network Operations and Management Seminar NOMS 2000,* Honolulu, Mai,2000.

[ROY 1995] ROY,S.”Discovering Rules for Fault Management”. In *Application Development and Management Strategies*, Gartner Group, Research Note K-560-1114, Jan.

[SCHWARTZ 2000]SCHWARTZ, S. H, ZAGER D.”Value-Oriented Network Management”. In *IEEE/IFIP Network Operations and Management Seminar NOMS 2000*, Honolulu, Mai,2000.

[SHEN 2000] SHEN, Dongxu, HELLERSTEIN, Joseph. “Predictive Models for Proactive Network Management: Application to a Production Web Server”. In *IEEE/IFIP Network Operations and Management Seminar NOMS 2000*, Honolulu, Mai,2000.

[STALLINGS 1999] STALLINGS,William. *SNMP, SNMPv2, SNMPv3, and RMON1 and 2*.Massachusetts: Addison Wesley, c1999. 619p.

[TAN 2000] TAN,Ming, LEE,Johnson, XU,Hao [et al]. "Wireless Usage Analysis for Capacity Planning and Beyond:A Data Warehouse Approach". In *IEEE/IFIP Network Operations and Management Seminar NOMS 2000,* Honolulu, Mai,2000.

[THOTTAN 1999] THOTTAN, M., JI, C. "Fault Prediction at the Network Layer using Intelligent Agents". In *Sixth IFIP/IEEE International Symposium on Integrated Network Management IM'99,* Boston, Mai,1999.

[TOMCAT 2002] TOMCAT. *"The Apache Jacarta Project".* Produced by TOMCAT. Disponível em <*http://jakarta.apache.org/tomcat/*>**.** Acesso em: 20/12/2002 .

[VAN HEMMEN 2000] VAN HEMMEN, L.J.G.T."Models Supporting The Network Management Organization". *International Journal of Network Management*; v.10, n. 6, p299-314, Nov-Dez 2000.

[YEMINI 1996] YEMINI, Shaula Alexander, KLIGER, Shmuel, MOZES, Eyal [et al]. "High Speed and Robust Event Correlation". *IEEE Communications Magazine*, v.34, n. 5, p82-90, Mai 1996.

[YUCEL 1999] YUCEL, Sakir, ANEROUSIS, Nikos. "Event Aggregation and Distribution in Web-based Management Systems". In *Sixth IFIP/IEEE International Symposium on Integrated Network Management IM'99,* Boston, Mai,1999.