CENTRO UNIVERSITÁRIO FEEVALE

RENÊ DE ASSIS DOS SANTOS

CORRELAÇÃO DE EVENTOS DE SISTEMAS DE GERENCIAMENTO DE TELECOMUNICAÇÕES UTILIZANDO RACIOCÍNIO BASEADO EM CASOS

Novo Hamburgo, Novembro de 2009.

RENÊ DE ASSIS DOS SANTOS

CORRELAÇÃO DE EVENTOS DE SISTEMAS DE GERENCIAMENTO DE TELECOMUNICAÇÕES UTILIZANDO RACIOCÍNIO BASEADO EM CASOS

Centro Universitário Feevale
Instituto de Ciências Exatas e Tecnológicas
Curso de Sistemas de Informação
Trabalho de Conclusão de Curso

Professor Orientador: Ricardo Ferreira de Oliveira

Novo Hamburgo, Novembro de 2009.

AGRADECIMENTOS

Gostaria de agradecer a todos os que, de alguma maneira, contribuíram para a realização desse trabalho de conclusão, minha família, colegas, amigos e professores.

Com certeza, sem o apoio decisivo nos momentos mais difíceis eu não teria conseguido chegar até aqui.

RESUMO

Os sistemas de gerenciamento de redes de telecomunicações, que consolidam os *logs* de funcionamento de e equipamentos de telecomunicações. Esses *logs* podem ser utilizados no monitoramento e diagnóstico de funcionamento de diversos sistemas. Usuários dos sistemas de gerenciamento analisam tais registros, na tentativa de se diagnosticar os problemas que podem surgir. Porém, além de entediante, a tarefa de análise de uma grande quantidade de logs pode se tornar infrutífera. Uma tentativa de agilizar o diagnóstico consiste na utilização de Raciocínio Baseado em Casos (RBC), uma técnica de Inteligência Artificial relativamente sedimentada, flexível e que se revela poderosa na medida em que recupera e consolida informações através de experiências passadas. Dessa forma, busca-se o estudo e modelagem de uma ferramenta baseada em RBC que contemple o diagnóstico de problemas utilizando como base os *logs* de eventos ocorridos numa rede de telecomunicações.

Palavras-chaves: Inteligência artificial. Raciocínio Baseado em Casos. Sistema de Gerenciamento de Redes. Eventos de sistema. *Log*.

ABSTRACT

The enterprises that provide telecommunication services use computerized systems to monitor their equipments networks. Such systems should provide mechanisms to locate and refrain the network's fails. Logs can be used in the diagnostic and monitoring of several systems, between them and the Telecommunications Network Management Systems. Users of management systems analyze such records, in an attempt to diagnose the problems that could arise. But, beyond boring, the task of analysis of a huge quantity of logs can become useless. An attempt to improve the diagnostics consists in the utilization of Case Based Reasoning (CBR), an Artificial Intelligence technique relatively uncomplicated, flexible and that reveal to be powerful in that it recovers and consolidates informations from past experiences. Thus, search the study and development of a tool based on CBR that includes the diagnosis of problems based on using the logs of events in a telecommunications network.

Key words: Case Based Reasoning. Artificial Intelligence. Telecommunications Management Network System. System events. *Log*.

LISTA DE FIGURAS

LISTA DE QUADROS

LISTA DE TABELAS

LISTA DE ABREVIATURAS E SIGLAS

CBR	*Case Based Reasoning*
GB	*Gigabyte*
CMIP	*Common Management Information Protocol*
CMIS	*Content Management Interoperability Services*
CMOT	*CMIP/CMIS over TCP/IP*
DAO	*Data Access Object*
GRT	Gerenciamento de Redes de Telecomunicações
IA	Inteligência Artificial
IEEE	*Institute of Electrical and Electronics Engineers*
IETF	*Internet Engineering Task Force*
ISO	*International Organization for Standardization*
ITU	*International Telecommunication Union*
MB	*Megabyte*
OMG	*Object Management Group*
OO	*Object Oriented* ou Orientado a Objetos
OSI	*Open Systems Interconnection*
QdS	Qualidade do Serviço
RBC	Raciocínio Baseado em Casos
RN	Regra de Negócio
ROM	*Read Only Memory*

SAO	Sistema de Apoio Operacional
SE	Sistema Especialista
SGRT	Sistema de Gerenciamento de Redes de Telecomunicações
SNMP	*Simple Network Management Protocol*
TB	*TeraByte*
TI	Tecnologia da Informação
TCP	*Transmission Control Protocol*
TCP/IP	*Transmission Control Protocol / Internet Protocol*
UDP	*User Datagram Protocol*
UML	*Unified Modeling Language*

SUMÁRIO

INTRODUÇÃO

A comunicação é uma das atividades mais primitivas do homem. É uma necessidade para a socialização e convívio humanos. Pode-se dizer que a comunicação é uma capacidade de enorme importância para a própria evolução humana. Segundo Shannon (1949), o problema fundamental da comunicação é o de reproduzir em um determinado ponto ou exatamente, ou aproximadamente, uma mensagem selecionada em outro ponto.

Conforme a história nos mostra, os meios de comunicação foram evoluindo em capacidade e alcance ao longo dos séculos. Desde as pinturas nas cavernas, até a era da internet em que vivemos. De acordo com a visão de Probst et al. (2002), a revolução na tecnologia das comunicações trouxe mudanças econômicas que acentuam a importância do conhecimento. Nos últimos anos, as redes de telecomunicações cresceram de forma exponencial, permitindo a circulação de conteúdo como vídeo, áudio e dados para todas as metrópoles mundiais e alcançando as localidades mais remotas.

Para alcançar esse patamar, muitas infra-estruturas foram construídas, baseando-se de várias tecnologias diferentes, cada uma sendo a mais adequada ao seu tempo. Cada avanço tecnológico na área que trazia significativos benefícios era adotado, na forma de novos produtos, novas redes, interligando de forma ainda mais rápida e dinâmica as populações.

Bernstein (1999) relata que o setor de telefonia norte americano enfrentou uma crise de serviço em 1970. As linhas telefônicas não poderiam ser instaladas em um prazo adequado e confiabilidade era um problema. Os sintomas foram pela primeira vez aparentes no sistema telefônico de Nova Iorque porque esse sistema enfrentou os graves desafios de atender às

necessidades de telecomunicações de centros financeiros do mundo. Logo ficou claro que estas complexidades de provisionamento foi um prenúncio de problemas. Foram criados sistemas informatizados e grupos de trabalho centralizados para gerenciar a rede de evitar uma crise nacional nos Estados Unidos. A central automática de relatórios de conexões foi entre a primeira era de sistemas de operações, que substituiu a configuração manual de conexões em mesa de teste de linha. Os resultados do teste foram notificados para os testadores que os analisaram e coordenou o trabalho dos técnicos no campo para isolar e corrigir problemas.

O crescimento exorbitante na área de telecomunicações tem aumentado a complexidade dos equipamentos utilizados. E o constante crescimento das redes incrementa a complexidade das próprias redes, que convergem tecnologias recentes com equipamentos projetados e em uso a décadas.

No atual contexto, organizações de grande porte suprem a necessidade de telecomunicações das populações, assumindo a responsabilidade de prover um serviço de grande importância, distribuído por extensões continentais, com altos níveis de disponibilidade e ainda com baixo custo operacional.

Para possibilitar a execução desses serviços, pode ser utilizado um Sistema de Apoio Operacional (SAO), portanto facilitando a tarefa gerenciar os equipamentos de telecomunicação que se encontram distribuídos geograficamente. Esses são os Sistemas de Gerenciamento de Redes de Telecomunicações (SGRT).

Carvalho (2003) explica que o gerenciamento de rede consiste em uma coleção de atividades requeridas para manter dinamicamente o nível de serviço em uma rede ou conjunto de redes. Estas atividades asseguram alta disponibilidade de recursos pelo rápido reconhecimento de problemas e degradação de performance, disparando funções de controle quando for necessário.

Uma das principais atribuições dos SGRT é a gerência de falhas, que segundo a norma ITU-T M.3400, delega ao SGRT a tarefa de monitorar os equipamentos que formam a

rede de telecomunicações na tentativa de antecipar possíveis falhas. Porém, na ocorrência de uma eventual falha em equipamento ou serviço, o SGRT deverá prover mecanismos para facilitar localização e diagnóstico do problema.

De acordo com Carvalho et al. (2003), para verificar se o nível de serviço atual corresponde ao desejado, informações são extraídas da rede para obter a configuração e performance em tempo real. As informações são obtidas continuamente ou sob demanda e armazenadas no base de dados da gerência da rede. Partes destes dados são submetidos à análise e outros dados são utilizados para comparar o estado real da rede com aquele desejado (planejado), permitindo verificar se alguma anomalia está ocorrendo. Deve-se preparar uma série de atividades para resolução de problemas, desde uma simples substituição de um dispositivo defeituoso até a execução de ferramentas mais sofisticadas para um diagnóstico mais acurado do problema.

Uma forma amplamente utilizada por diversas gerações de equipamentos e sistemas para propiciarem o monitoramento e histórico de falhas é a geração de *logs*. Dessa forma, são gerados inúmeros registros do que acontece por dentro dos equipamentos, e esses dados são transmitidos ao SGRT para serem consolidados e apresentados aos usuários.

Nessas condições, diante de um problema na rede, o usuário é informado que algo de anormal está acontecendo em alguma parte da rede. Para solucionar o problema é necessário diagnosticar a causa raiz do mau-funcionamento. O que pode ser uma tarefa entediante e infrutífera devido ao grande volume de dados captados pelo SGRT.

Considerando que os problemas que ocorrem numa rede de telecomunicações geralmente são recorrentes, e as conseqüências desses problemas ficam armazenados num repositório unificado como histórico de *logs* de um SGRT, verifica-se a possibilidade de aplicar alguma técnica de análise de dados para tentar auxiliar o diagnóstico de problemas futuros.

Geralmente pode-se recorrer a técnicas manuais de análise de dados, ou ferramentas de mineração de dados, porém essas técnicas podem não serem tão eficazes na medida em que

os dados registrados podem não estar bem estruturados, ou ainda possuir volume e/ou granularidade muito grandes.

Deve-se considerar que os dados captados pelo sistema podem prover de equipamentos muito distintos, que utilizam tecnologias e tempos de resposta diferentes, e são provenientes de diversos fabricantes.

Uma forma de possibilitar a análise de grandes quantidades de *logs* consiste na correlação destes registros. Dessa forma, um conjunto de notificações são agrupadas segundo algum critério, de forma a facilitar o entendimento do que está acontecendo na rede, ou em determinado equipamento.

Kay (2003) explica que uma variedade de técnicas e operações são associadas com correlação de eventos, tais como: compressão, em que registros duplicados são omitidos; contagem, em que a partir de um número de notificações similares são relatados apenas um; generalização, em que apenas alarmes mais importantes são reportados; e correlação baseada em tempo, em que eventos são associados uns aos outros de acordo com o intervalo de tempo em que eles ocorrem.

Uma forma de utilizar sistemas informatizados na resolução de problemas consiste na utilização de Inteligência Artificial (IA). Diferentes técnicas de IA podem ser utilizadas em correlação de eventos, com bom potencial de auxílio de diagnóstico para resolução de problemas. Numa planta de equipamentos que formam uma rede de telecomunicações geralmente são utilizados muitos equipamentos de mesmo modelo, o que por si só torna as notificações de problemas recorrentes muito semelhantes.

Uma técnica de IA que tem sido utilizada com bons resultados é o Raciocínio Baseado em Casos, que além de possuir foco no diagnóstico de casos recorrentes, traz vantagens como possibilidade de aprender conforme novos problemas são catalogados em sua base de dados, o que é feito pelos próprios usuários finais, o que facilita a manutenção e adaptação do sistema a novos cenários.

Este trabalho tem como objetivo demonstrar a modelagem de um sistema de correlação de eventos, utilizando raciocínio baseado em casos. Para tal, serão relatados de uma forma geral em que consistem os Sistemas de Gerenciamento de Redes, com ênfase na gerência de falhas; serão listados os conceitos de Inteligência Artificial com destaque ao Raciocínio Baseado em Casos e seu funcionamento. Baseando-se nesses conceitos, e no que a análise de sistemas presume, será modelado um sistema de correlação de eventos utilizando raciocínio baseado em casos.

Este trabalho está dividido em quatro capítulos. No capítulo 1, são tratados os principais assuntos referentes ao gerenciamento de redes e sua importância na manutenção de redes de computadores. No capítulo 2 é abordado a Inteligência Artificial com ênfase no Raciocínio Baseado em Casos, seus aspectos e sua importância como ferramenta na solução de problemas e busca de informação. No capítulo 3 é apresentada a problemática em questão e proposta de ferramenta para resolver o problema.

Por fim, o capítulo 4 terá a função de demonstrar uma modelagem de Sistema de Correlação de Eventos capaz de extrair informações úteis a partir de um conjunto de dados de eventos consolidados por um SGRT.

1 GERENCIAMENTO DE REDES DE TELECOMUNICAÇÕES

De acordo com Farrel et al. (2009), em algum nível todos equipamentos de rede necessitam de algum gerenciamento. Até mesmo os dispositivos mais simples possuem necessidades de gerenciamento físico de mesma forma que são necessitam energia para funcionar. E alguns equipamentos necessitam de alguma forma de configuração para "dizer-los" qual papel eles irão cumprir na rede e precisamente como proceder.

Neste capítulo serão abordados conceitos e temas a respeito da área de Gerenciamento de Redes de Telecomunicações (GRT).

1.1 História do Gerenciamento de Redes de Telecomunicações

Tanenbaum (1997) explica que no início da operação da ARPANET, se a demora no acesso a um *host* se tornasse grande, quem detectasse o problema simplesmente executaria o programa Ping para tirar o pacote de seu destino. (...) Quando a ARPANET se transformou na rede mundial Internet, com diversos *backbones* e operadores, essa solução deixou de ser adequada, e houve a necessidade de se criarem melhores ferramentas para o gerenciamento da rede.

Tanenbaum (1997) descreve que em maio de 1990 a RFC 1157 foi publicada, definindo a versão 1 do SNMP (*Simple Network Management Protocol*), junto com um documento complementar (a RFC 1155), que trata de informações de gerenciamento.

Segundo Cyclades (2000), o desenvolvimento de padrões para GRT iniciou-se a partir da segunda metade da década de 80, com um trabalho conjunto da comunidade Internet

(Internet Engineering Task Force – IETF) e da comunidade ISO. Esse trabalho analisou o gerenciamento de redes sob dois aspectos: o desenvolvimento de protocolos de comunicação e o desenvolvimento de um sistema para organizar as informações a ser gerenciadas.

Desse esforço, originaram-se dois protocolos: SNMP e *CMIP over TCP/IP ou* CMOT.

Farrel et al. (2009) explica que atualmente, os dois principais protocolos de gerenciamento de redes são o SNMP e CMIP, sendo que CMIP inclui CMIP sobre TCP/IP (CMOT). Esses protocolos provêem mecanismos para obter, alterar e transportar dados de gerenciamento de rede pela rede.

Tabela 1.1 – Protocolos de gerenciamento iniciais

Protocolo	Características
SNMP	Protocolo mais simples, considerado no início uma solução de curto prazo para a necessidade de gerenciamento de redes TCP/IP
CMOT	A solução de longo prazo, abrangendo também a estrutura de gerenciamento OSI, para qualquer tipo de rede

Fonte: (Cyclades, 2000).

1.2 O protocolo SNMP

O Simple Network Management Protocol (SNMP) é um protocolo de rede baseado em UDP (RFC 3411). É usado mais extensivamente em sistemas de gerenciamento de rede para monitorar os dispositivos conectados em rede para as condições que merecem uma atenção administrativa. O SNMP é um componente do *Internet Protocol Suite*, tal como definido pela Internet Engineering Task Force (IETF). Se constitui de um conjunto de normas para a gestão de rede, incluindo um protocolo de camada de aplicação, um modelo de dados, e um conjunto de objetos de dados.

Segundo Tanenbaum (1997), UDP é um protocolo sem conexão não confiável para aplicações que não necessitam nem de controle de fluxo, nem da manutenção da seqüência das mensagens enviadas. Ele é amplamente utilizado em aplicações em que a entrega imediata é mais importante do que a entrega precisa, como a transmissão de voz ou vídeo.

O protocolo SNMP faz parte da quinta camada do protocolo TCP/IP, também conhecida como camada de aplicação. Tanenbaum (1997) explica que a camada de aplicação contém protocolos de alto nível, como por exemplo o protocolo de terminal virtual (TELNET), protocolo de transferência de arquivos (FTP), e o protocolo de correio eletrônico (SMTP). O protocolo de terminal virtual permite que um usuário de um computador estabeleça *login* em uma máquina remota e trabalhe nela. O protocolo de transferência de arquivos permite mover dados com eficiência de uma máquina para outra. Originalmente o correio eletrônico era um tipo de transferência de arquivos; no entanto, posteriormente um protocolo especializado foi desenvolvido para essa função. Muitos protocolos foram incluídos no decorrer dos anos, como o DNS (*Domain Name Service*), que mapeia os nomes de *host* para seus respectivos endereços de rede, o NNTP, o protocolo usado para mover novos artigos, e o HTTP, o protocolo usado para buscar páginas na WWW (*World Wide Web*), entre outros.

Segundo Hewlett Packard (2000b), o gerenciamento de redes de computadores requer uma abordagem que simplifique os problemas potencialmente complexos de comunicação e coordenação. A abordagem prevalecente, que foi adotada pelo protocolo SNMP, é tratar a rede como uma coleção de entidades comunicativas e cooperativas. Dessa forma, existindo dois tipos básicos de entidades: processos gerenciadores (gerentes), e processos gerenciados (agentes).

Conforme Tanenbaum (1997), os nós gerenciados podem ser *hosts*, roteadores, pontes, impressoras ou qualquer outro dispositivo capaz de comunicar informações de estado para o mundo externo.

Miller (1997) explica que um gerente provê a interface entre o gerente de rede humano e os equipamentos sendo gerenciados. O gerente também provê o processo de gerenciamento da rede. O processo de gerenciamento da rede efetua tarefas como medir tráfego num segmento remoto de rede ou gravar a velocidade de transmissão e o endereço físico de uma interface de rede de um roteador. O gerente também inclui algum tipo de saída de dados, geralmente gráfica, para exibir dados de gerenciamento, estatísticas históricas, entre outros. Um exemplo comum de exibição gráfica é um mapa da topologia da rede, mostrando

as localizações dos segmentos da rede; selecionar um segmento em específico deve exibir seu estado operacional atual.

Hewlett Packard (2000b) define um gerente como um processo ou nó de rede que é um participante ativo no gerenciamento da rede. O gerente solicita e interpreta dados sobre os equipamentos da rede e tráfego de rede, e tipicamente interage com um usuário para atender as intenções do usuário. Portanto, um gerente pode também engatilhar mudanças em um agente por meio da mudança do valor de uma variável no nó de rede do agente. Gerentes são tipicamente implementados como Sistemas de Gerenciamento de Redes.

Carvalho et al. (2002) o modelo tradicional de um SGRT consiste de uma aplicação Gerente requisitando informações dos Agentes (equipamentos gerenciados). Em muitos casos os Agentes também tomam a iniciativa de se comunicar com o Gerente, enviando alarmes (denominados *traps*) sobre um evento anormal (queda de link, tentativas de acesso forçadas com senhas erradas etc).

Sob a perspectiva do gerenciamento de redes, um agente geralmente é uma entidade passiva. O agente responde às requisições dos gerentes, fornecendo e alterando os valores de variáveis locais quando necessário. Um agente pode se tornar uma entidade ativa ao emitir mensagens não solicitadas chamadas “notificações” ou *traps* para alertar gerentes de eventos locais significantes.

O agente encarrega-se da atividade SNMP no nó de rede gerenciado. Ele pode ser simples ou complexo, dependendo do dispositivo que ele representa. Assim como um gerente, um agente pode ser um de muitos processos em um computador específico. De outra forma, ele pode ser implementado na memória ROM de um equipamento de rede como um *hub* ou Modem.

Um equipamento que não suporta diretamente o protocolo SNMP é dito como “estrangeiro”. Um *proxy* é um agente que traduz entre SNMP e o protocolo privado de um equipamento estrangeiro.

Segundo Bernstein (1999), um agente *proxy* com interface SNMP pode ser usado em elementos de rede que não possuem agentes integrados quando o tempo é curto ou os recursos para desenvolvimento são escassos.

Gerentes não precisam conhecer qualquer detalhe interno sobre o objeto gerenciado por um agente. Igualmente, um agente SNMP pode responder requisições SNMP para qualquer gerente SNMP. O agente não precisa conhecer o contexto da requisição, ou a estrutura do gerente efetuando a requisição. O agente simplesmente valida a requisição, responde ela, e entra em seu estado passivo aguardando a próxima requisição.

Segundo Hewlett Packard (2000b), esta divisão de responsabilidades simplifica as soluções de gerenciamento de redes. O conceito de gerentes e agentes provê uma solução genérica para problemas outrora complexos. Todos os equipamentos compartilham um modelo comum de comunicação, que é o modelo proposto no protocolo SNMP.

Segundo Cyclades (2000) o modelo simplificado de funcionamento do protocolo SNMP pode ser representado pela figura abaixo.

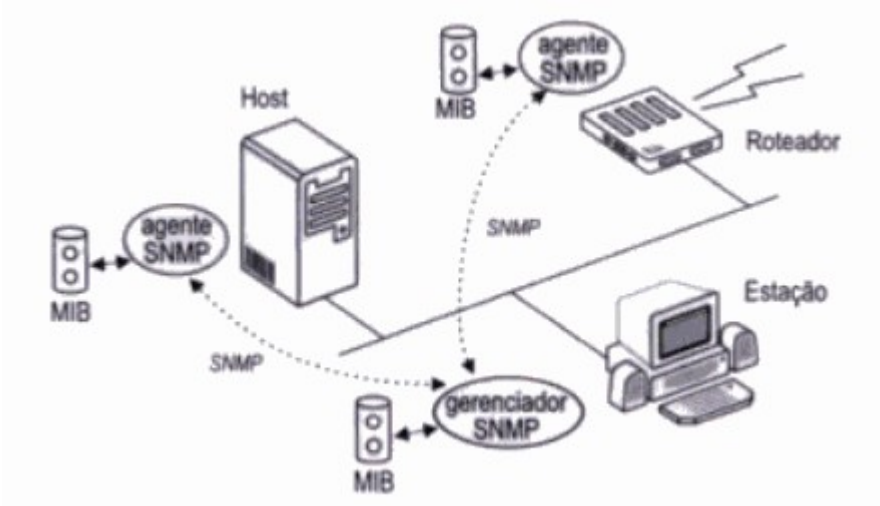

Figura 1.1 – Modelo de gerenciamento SNMP

Fonte: (Cyclades, 2000).

A função de cada componente pode ser resumida na seguinte tabela:

Tabela 1.2 – Resumo das funcionalidades de cada componente do protocolo SNMP

Componente	Descrição
Agente SNMP	Programa executado nas entidades a ser gerenciadas (*hosts, hubs, roteadores*). Tem como função principal a de responder a solicitações de infirmações feitas pelo programa gerenciador da rede.
Gerenciador SNMP	Programa executado em um equipamento que faz a conversão de solicitações do usuário responsável pelo gerenciamento da rede em ações e consultas aos gerentes por ele gerenciados. Esse equipamento onde está ativo um gerenciador é por isso mesmo denominado *estação de gerenciamento* ou *console de gerenciamento.*
Protocolo SNMP	Protocolo de comunicação responsável pela troca de mensagens entre os programas gerenciadores e os agentes.

Fonte: (Cyclades, 2000).

O protocolo SNMP tem sido utilizado desde seu surgimento, porém o que foi previsto com o protocolo CMOT não tem acontecido. Quando fabricantes buscaram um protocolo com mais funcionalidades, permitindo uma comunicação mais robusta com os elementos gerenciados, foram utilizados protocolos próprios ou ainda outros protocolos diversos.

Dessa forma, o protocolo SNMP é utilizado até hoje, pois apesar de simples, ele provê um protocolo comum entre uma grande variedade de equipamentos de diferentes fabricantes.

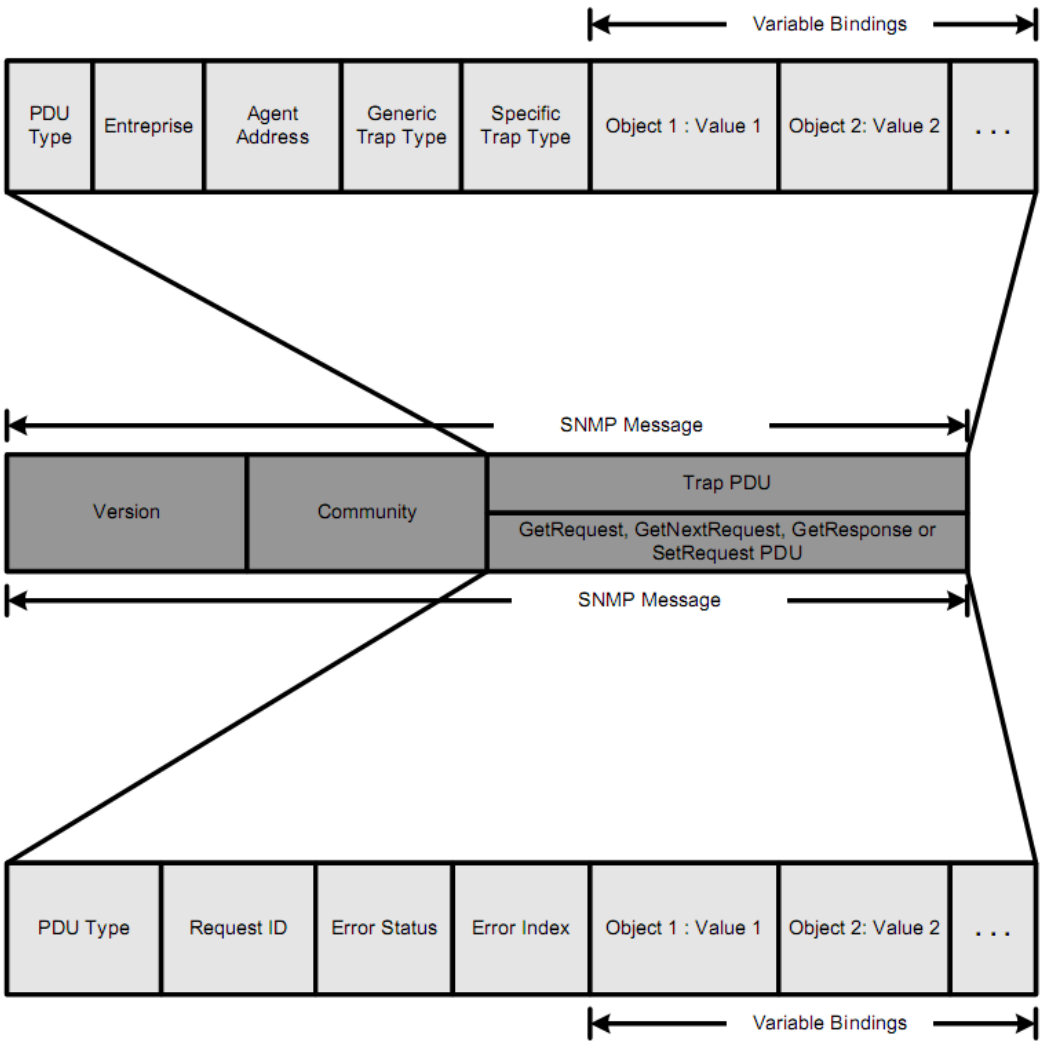

Figura 1.2 – Formato de uma mensagem SNMP

Fonte: (Infrax, 2006)

1.3 Sistema de Gerenciamento de Redes de Telecomunicações

As redes devem ser gerenciadas por vários motivos, explica Farrel et al. (2009). A partir de uma perspectiva inteiramente prática, equipamentos devem ser monitorados para certificar que eles estão funcionando corretamente.

Serviços que são oferecidos por uma rede, também precisam sem gerenciados, particularmente quando estes estão aprovisionados. Nesses casos equipamentos são acessados e configurados. Serviços gerenciados também requerem monitoramento e manutenção. Por exemplo, se um provedor de serviço oferece um serviço de rede privada virtual a um cliente,

pode ser necessário monitorar a saúde e desempenho de segmentos da rede que suportam o tráfego daquele cliente, para certificar que ele está obtendo o serviço o qual está pagando. Na verdade, esse monitoramento é muitas vezes uma necessidade contratual.

Nesses cenários, é extremamente difícil ou quase impossível para operadores de médias a grandes redes monitorar cada equipamento em suas redes manualmente. Ao invés disso, muitos preferem fazê-lo de uma maneira automatizada.

Segundo Farrel et al. (2009) o processo de monitoramento envolve coletar dados sobre os dispositivos desejados, processar algum ou todos os dados obtidos, exibir os dados (processados), e arquivar um resumo desses dados.

A norma ITU-T M.3010 expõe que um SGRT deve prover funcionalidades de gerenciamento às várias peças da rede de telecomunicações. Deve ainda oferecer funcionalidades de gerenciamento a outros sistemas de forma a permitir o suporte completo de redes de telecomunicações nacionais e internacionais. Alguns alcançam seu gerenciamento usando a abordagem centralizada, utilizando sofisticados centros de operação. Contudo, outros podem escolher ter vários pequenos centros de operação que são distribuídos. Nesse caso, é extremamente demorado, portanto custoso, para um operador manualmente conectar cada console de equipamento para monitorar seu estado, isolar falhas, ou configurar o dispositivo. Isso se torna mais óbvio quando se considera aqueles provedores de redes onde os dispositivos de rede estão localizados sobre uma ampla gama de áreas geográficas. Neste caso, torna-se ainda mais caro para viajar para um local remoto ou contratar pessoal adicional para estar no local onde esses dispositivos suplementares estão localizados.

O princípio de manter o SGRT logicamente distinto das redes e serviços sendo gerenciados introduz o prospecto de distribuir as funcionalidades do SGRT para implementações centralizadas ou descentralizadas. Isso significa que de um número de SGRT, usuários podem efetuar o gerenciamento de uma ampla variedade de equipamentos, redes e serviços distribuídos.

Segurança e integridade de dados distribuídos são fundamentais requisitos para a definição de uma arquitetura genérica. Um SGRT deve permitir acesso e controle a partir de fontes consideradas fora da rede gerenciada. Mecanismos de segurança podem ser necessários em vários níveis.

Hewlett Packard (2000a) propõe que no gerenciamento de redes, sistemas ou serviços, um usuário tipicamente utilizará uma das três abordagens a serem descritas. A escolha da abordagem dependerá do trabalho do usuário e a forma que o trabalho está estruturado na sua organização. O usuário também poderá utilizar mais de uma abordagem dependendo dos requerimentos do trabalho que ele está tentando cumprir.

- Abordagem orientada a exceções
- Abordagem baseada em tarefas
- Abordagem orientada ao usuário

Na abordagem orientada a exceções o usuário deve possuir um bilhete de problema no sistema, navegador de mensagens ou representação gráfica que o notifica de problemas em seu domínio de gerenciamento. Ele recebe uma notificação, para então começar a tarefa de investigação, diagnóstico, solução e documentação do problema que causou a notificação.

Exceções nem sempre são problemas que ocorrem no atual ambiente de rede, sistema ou serviços; as exceções podem ser requisições para novos serviços ou para equipamentos a serem adicionados ao ambiente. Algumas exceções podem ser tratadas por ações automáticas pré-definidas ou por uma seqüência de passos para orientar o usuário.

Uma tarefa consiste na execução de uma parte de um trabalho. Tarefas podem ser definidas em múltiplos níveis, ou seja, tarefas podem ser compostas de múltiplas subtarefas ou tarefas simples que definem uma unidade de execução. Tarefas podem envolver a manipulação de um único objeto ou múltiplos objetos. O gerenciamento de rede, sistemas e serviços geralmente envolvem tarefas bem definidas que devem ser executadas ou baseadas em periodicidade ou baseadas em eventos. Na abordagem baseada em tarefas, um usuário executa um conjunto de passos necessários para completar aquela tarefa. Para tarefas que

serão repetidas por um número de vezes por diferentes usuários ou pelo mesmo usuário, é possível definir uma seqüencialidade otimizada dos passos das tarefas para alcançar o melhor desempenho dos usuários.

Hewlett Packard (2000a) explica que a abordagem baseada em tarefas é frequentemente usada ao configurar um novo equipamento, ao fazer mudanças em configurações, na resolução de problemas rotineiros ou ao executar trabalhos de rotina.

Devido ao fato de que os problemas complexos em redes, sistemas e serviços podem existir e os grupos de telecomunicações e tecnologia da informação possuírem variedades de produtos a serem gerenciados, existe a necessidade de acesso irrestrito à informação e funcionalidades. Esta abordagem orientada ao usuário permite máxima liberdade para um usuário, através da habilitação ao acesso a múltiplas ferramentas e capacidades com restrições limitadas.

A abordagem orientada ao usuário é freqüentemente utilizada no planejamento de modificações para uma rede e infraestrutura de sistema, na verificação das condições de um objeto específico ou um conjunto de objetos; e na resolução de problemas complexos que não foram previamente encontrados ou previstos.

Para atender os objetivos listados, a norma ITU-T M.3010 define cinco funcionalidades de gerenciamento em que os SGRT podem atuar:

- Gerenciamento de desempenho;
- Gerenciamento de falhas;
- Gerenciamento de configuração;
- Gerenciamento de contabilidade;
- Gerenciamento de segurança.

Bernstein (1999) resume que o gerenciamento de configuração inclui o inventário e conectividade dos elementos da rede. Gerenciamento de falhas busca a redução da inatividade

da rede. Gerenciamento de desempenho monitora dados filtrados rede que extrapolam os limiares para indicar a saúde da rede. Gerenciamento de contabilidade prevê retorno financeiro e captura os custos da rede. Finalmente, o gerenciamento da segurança é vital para proteger o investimento de hardware e software, bem como a integridade da rede e dados, contra invasão, pirataria, ou alteração não autorizada.

1.3.1 Gerenciamento de desempenho

A norma ITU-T M.3400 propõe que o gerenciamento de desempenho tem por finalidade prover funcionalidades para avaliar e reportar desde o comportamento de um equipamento de telecomunicação até mesmo a eficiência da rede como um todo. Essa função é obtida reunido e analisando dados estatísticos tendo como propósito monitorar e corrigir o comportamento e eficiência da rede ou equipamento de rede; e ajudar a planejar, aprovisionar, manter e medir a qualidade.

De acordo com Miller (1997) o gerenciamento de desempenho garante que o administrador satisfaça as necessidades dos usuários finais em todos os momentos. Para fazer isso, o administrador deve selecionar os sistemas de hardware e software de acordo com as necessidades do conjunto de redes, e então testar esses sistemas em seu potencial máximo. Desempenho e gerenciamento de falhas estão intimamente relacionados, já que é necessário para eliminar, ou pelo menos minimizar, as falhas para obter um ótimo desempenho. Muitas ferramentas estão disponíveis para medir o desempenho. Estes incluem os analisadores de protocolo, software de monitoramento de rede e vários utilitários que vêm com o console de programas de sistemas operacionais de rede.

Segundo Farrel et al. (2009), o ponto de partida para a gerência de desempenho é a garantia da qualidade de serviço. Qualidade de serviço é um mecanismo típico para a transmissão de informação de interface entre fornecedor (isto é, a um responsável por uma rede de comunicações ou para a infra-estrutura de TI) e do cliente, assim o usuário do serviço. Sua importância aumenta à medida que mais as relações cliente-fornecedor estão envolvidos na implementação de redes corporativas e sistemas distribuídos.

O SGRT coleta dados de qualidade do serviço (QdS) a partir dos elementos de rede e auxilia melhorias na QdS. O SGRT pode requisitar relatórios de dados de QdS aos elementos de rede, ou um relatório pode ser enviado automaticamente de acordo com agendamento e/ou limiares de exceção. Relatórios de QdS oriundos de elementos de rede podem consistir de dados brutos que podem ser analisados externamente ao elemento de rede, ou o elemento de rede pode ser capaz de se encarregar de parte da análise antes de reportar os dados.

1.3.2 Gerenciamento de falhas

Segundo a norma ITU-T M.3400 gerenciamento de falhas é uma funcionalidade exercida por um SGRT que engloba a detecção, isolamento e correção de operação anormal da rede de telecomunicações e seu ambiente, provendo facilidades para a execução das fases de manutenção previstas. Ainda segundo a ITU-T M.3400 o gerenciamento de falhas inclui medir os níveis de qualidade, incluindo medições de confiabilidade, disponibilidade e durabilidade.

Farrel et al. (2009) adiciona que, comparando com sistemas não conectados, gerenciamento de falhas em redes de computadores e sistemas distribuídos é mais difícil devido a muitas razões. Estas incluem o grande número de componentes envolvidos, a ampla distribuição física dos recursos, a heterogeneidade dos componentes de hardware e software, e os diferentes domínios abrangidos (por exemplo, o pessoal de diferentes unidades organizacionais).

Miller (1997) esclarece que primeiro é necessário identificar, depois reparar falhas de rede. Existem duas formas de gerenciar falhas: reativamente ou proativamente. A gerência reativa espera um problema e então trabalha para resolvê-lo. Uma gerência proativa examina a rede se estão sendo excedidos limiares operacionais críticos, tais como taxa de utilização da rede. Se excessos ocorrem, o administrador proativo determina suas fontes e reduz-os de acordo.

Uma funcionalidade importante da gerência de falhas é a localização de falhas. Onde a informação inicial é insuficiente para a localização de falhas deve-se acrescentar informação

oriunda de rotinas adicionais de localização de falhas. As rotinas podem empregar sistemas de testes internos ou externos e podem ser controlados por um SGRT.

O processo de determinar falhas começa com a informação recebida pelo gerenciamento de status da rede, ou seja, informação como relatório de alarme, mensagem do usuário oriunda de cliente, relatório de monitoramento em tempo real, ou resultados de testes baseados em rotinas de teste automáticas ou manuais. Baseado na informação de falha recebida, um *ticket* de problema é aberto um gerenciador de *tickets*, e consolidado com o *ticket* relatado.

O gerenciamento de evento analisará o relatório de falhas e existem quatro oportunidades de novo estágio: diagnosticado, ajuda requerida, suspenso ou retirou-se.

Após correto diagnóstico o problema será reparado, e o *ticket* de problema pode ser fechado. De outro modo, se o problema poderia ser re-selecionado para ser resolvido. Problemas que desaparecem ou estão suspensos, ou quando seu tempo expira, podem ser alterados para um estado “expirado”. Depois que o problema é determinado, a mensagem do problema é enviada ao cliente.

Após esse processo, a base de dados estará atualizada e criará um registro histórico. Em condições especiais o sistema desenvolve uma hipótese para resolver um problema.

A norma ITU-T M.3400 indica também a existência de armazenamento de notificações de eventos ocorridos na rede. Esse registro de operações não deve ser apagado, para que se mantenha o histórico de funcionamento da rede como um todo. Essa característica do SGRT pode ser explorada por ferramentas externas para auditoria, ou mesmo para analisar e encontrar problemas ocorridos e soluções empregadas.

1.3.3 Gerenciamento de configuração

Segundo Farrel et al. (2009), configuração é uma adaptação de sistemas para ambientes operacionais e inclui a instalação de novos softwares, expansão de software antigo,

anexar dispositivos, ou fazer alterações na topologia da rede ou a carga de tráfego. Apesar de configuração também englobar os aspectos da instalação física, é normalmente realizado através de um software de controle de geração e estabelecimento de parâmetros, que incluem parâmetros de função ou seleção; parâmetros de autorização; parâmetros de protocolo (comprimentos de mensagens, janelas, temporizadores, prioridades); parâmetros de fixação (tipo e classe de dispositivo, concorrência, taxa de *bits* de paridade); entradas em tabelas de roteamento, servidores de nomes, listas, bem como os parâmetros filtro de *bridges* (endereços, tipos de protocolos, integração); parâmetros de *spanning tree* para uma *bridge* (de prioridade de *bridge* ou de porta), e parâmetros para os caminhos de ligação dos roteadores (interfaces, a velocidade, o fluxo de procedimentos de controle), o tamanho máximo de arquivo, os tempos de computação e serviços permitidos.

Miller (1997) observa que as redes são sistemas dinâmicos, e os administradores de rede precisam deslocar pessoal e reorganizar as suas necessidades de processamento. Esse aspecto da gerência da rede pode ser tão simples como a reorganização conectores modulares em um hub de fiação, ou tão complexo como a instalação de uma rede e seus servidores, circuitos de comunicação, e assim por diante, em um local remoto. Portanto, um aspecto significativo da função de gerência de rede envolve rastrear todas estas alterações usando algum tipo de banco de dados.

No gerenciamento de configuração, o SGRT deve prover funcionalidades para exercer controle sobre, identificar, coletar e prover dados aos elementos de rede.

A norma ITU-T M.3400 estabelece que o gerenciamento de configuração preveja o seguinte conjunto de funcionalidades:

- Planejamento e engenharia da rede;
- Instalação;
- Planejamento de serviços e negociação;
- Aprovisionamento;
- *Status* e controle.

Planejamento e engenharia da rede lidam com as funcionalidades associadas à determinação da necessidade de crescimento em capacidade e a introdução de novas tecnologias. Ela envolve a avaliação de planos alternativos e a inserção de tais planos numa base de dados que apoiará o grupo de funcionalidades de aprovisionamento.

O SGRT pode auxiliar a instalação de equipamentos da rede de telecomunicação. Isso cobre tanto a extensão como a redução do sistema. Alguns elementos de rede buscam por troca de dados iniciais entre eles e SGRT. Outra funcionalidade é a instalação de programas em elementos de rede a partir de bases de dados dos SGRT. Adicionalmente, dados administrativos podem ser trocados entre os elementos de rede e o SGRT. Programas de testes de aceitação podem ser feitos sobre controle, ou auxiliados pelo SGRT.

Planejamento de serviços e negociação lida com planejamento para introdução de novos serviços e com contratos de clientes para estabelecer novos serviços, mudar características de serviços e desconectar serviços.

Aprovisionamento consiste de procedimentos que são necessários para colocar um equipamento em serviço, não incluindo instalação. Uma vez que a unidade está pronta para serviço, os programas de apoio são inicializados através do SGRT. Os estados da unidade tais como: em serviço, fora de serviço, ocioso, reservado e parâmetros selecionados podem ser controlados pelas funcionalidades de aprovisionamento.

O SGRT provê a capacidade de monitorar e controlar certos aspectos do elemento de rede sob demanda. Exemplos incluem verificar ou alterar o estado de serviço de um elemento de rede ou uma de suas sub-partes (em serviço, fora de serviço, ocioso) e inicializando testes de diagnóstico no elemento de rede. Normalmente, uma verificação de estado é provida em conjunto com cada funcionalidade de controle, na tentativa de verificar se a ação esperada ocorreu.

1.3.4 Gerenciamento de contabilidade

De acordo com Farrel et al (2009) as funções de gerenciamento de contabilidade inclui funções de gerenciamento do uso (utilização, edições de utilização e validação de

eventos de chamada ou solicitações de serviço, a correção de erros de uso, acumulação de uso, a correlação de uso, a agregação de uso, distribuição de uso); funções do processo de contabilidade (testes de uso, a fiscalização do uso, a administração da coleta de dados de utilização); funções de controle (administração tarifárias, controle de mudanças no sistema de tarifas, controle de geração de registro, controle de transferência de dados, controle de armazenamento de dados) e cobrança (geração de despesas, a produção de contas, processamento de pagamentos, cobrança de dívidas, reconciliações externas, processamento de contrato). Muitas das funções mencionadas são especialmente importantes para os prestadores públicos de telecomunicações. Em tais ambientes, os serviços são frequentemente serviços "multiredes" (isto é, os nós de rede, provedores diferentes, ou assinantes de serviços móveis podem estar envolvidos). Portanto, a gerência de contabilidade deve abordar a coleção distribuída de dados de uso, para geração de relatório (quase em tempo real), e múltiplas estratégias de tarifação.

Miller (1997) define contabilidade como lidar com pessoas reais, usando recursos de rede real com as despesas de funcionamento real. Exemplos destes custos incluem o uso do disco e arquivamento de dados, as despesas de telecomunicações para acesso a dados remotos, e os encargos para o envio de mensagens de correio eletrônico. Você também pode usar o gerenciamento de contabilidade para determinar se a utilização de recursos de rede está aumentando por causa do crescimento, o que pode indicar a necessidade de acréscimos ou ajustes no futuro próximo.

A norma ITU-T M.3400 define que o gerenciamento de contabilidade permite a medida de uso dos serviços da rede e a determinação dos custos operacionais e cobranças aos clientes que fazem o uso da rede. Ele também auxilia a determinação dos preços dos serviços.

1.3.5 Gerenciamento de segurança

Na norma ITU-T M.3400 é explicado que a segurança do gerenciamento é requisito de todas as áreas funcionais de gerenciamento, e para todas as transações do SGRT.

A funcionalidade de gerenciamento de segurança inclui serviços de segurança para comunicações e detecção de eventos de segurança e relatório.

Miller (1997) esclarece que a segurança protege a rede. Defende contra vírus, garante que os usuários remotos e locais são autenticados e instala sistemas de criptografia em todos os circuitos de comunicação que se conectar a um local remoto.

De acordo com Farrel et al. (2009) o termo gerenciamento de segurança não é usado para se referir à segurança do gerenciamento (ou seja, garantir o gerenciamento é realizada de forma segura), mas para o gerenciamento da segurança em sistemas distribuídos.

Farrel et al (2009) expõe que um aspecto importante são os recursos de uma empresa precisa proteger: informação, infra-estruturas de TI, serviços, e produção representam os valores que estão expostos a ameaças de ataque ou uso indevido. As medidas de segurança que refletem os resultados de análises de risco ou de ameaça de segurança são necessárias para evitar danos e perdas. Ameaças típicas são criadas por:

- Ataques passivos: espionagem de informações, criando um perfil de usuário ou uma análise de tráfego de dados indesejados ou roubo de informações (senhas, etc.)

- Ataques ativos: mascaramentos (isto é, os usuários fingindo ser outra pessoa, ou falsificação); dando prioridade para ou atrasando mensagens; modificar mensagens, manipulando recursos através da sobrecarga, reconfiguração, reprogramação e assim por diante (acesso não autorizado, vírus, cavalos de Tróia, ataques de negação de serviço).

- O mau funcionamento dos recursos.

- Comportamento defeituoso ou inadequado e funcionamento com resultado incorreto.

Os serviços de segurança provêm um conjunto de serviços de segurança para comunicações como definido na norma ITU-T X.800 para autenticação, controle de acesso, confidencialidade de dados e integridade de dados. Adicionalmente, um conjunto pervasivo

de mecanismos de segurança são definidos como aplicáveis a qualquer comunicação, como detecção de eventos, gerenciamento de traços de auditoria e recuperação de segurança.

2 INTELIGÊNCIA ARTIFICIAL E RACIOCÍNIO BASEADO EM CASOS

Neste capítulo são apresentados os conceitos referentes à tecnologia empregada deste trabalho: um sistema baseado em RBC para correlação de eventos de SGRT.

2.1 Introdução

Como demonstra Russel (2004), Inteligência Artificial (IA) possui algumas definições distintas, criadas de acordo com a abordagem utilizada. Dentre essas abordagens, nota-se uma tentativa de utilizar o comportamento do raciocínio humano.

Ainda segundo Russel (2004), o pensamento humano é estudado pela ciência cognitiva. Reunindo modelos computacionais da IA e técnicas experimentais da psicologia para tentar construir teorias precisas e verificáveis a respeito dos processos de funcionamento da mente humana.

2.2 Sistemas Especialistas

Os Sistemas Especialistas (SE) caracterizam-se pela sua capacidade em chegar a conclusões que somente seriam possíveis por pessoas que possuem o domínio de uma específica área de conhecimento. Os especialistas são consultados quando na concepção do sistema, dessa forma, armazenando o conhecimento no próprio SE.

Como afirma Luger (2004), uma característica fundamental dos SE é que sua estratégia para resolver problemas é dependente do conhecimento do especialista humano no domínio.

Conforme Silva (2008) a elaboração de um SE, envolve varias etapas, como:

- Aquisição do Conhecimento;
- Representação do Conhecimento;
- Motor de Inferência;
- Interface com Usuário;
- Aprendizagem;
- Justificativa.

Luger (2004) destaca algumas das dificuldades encontradas na utilização de SE:

- Dificuldade em capturar conhecimento "profundo" do domínio do problema. O SE possui conhecimento até onde foi programado, não aprenderá novos tipos de conhecimentos para os correlacionar.
- Falta de robustez e flexibilidade. Se num primeiro momento um sistema especialista não conseguir resolver um problema, uma revisão dos dados coletados não trará resultado adicional.
- Inabilidade de fornecer explanações aprofundadas. Os SEs são incapazes de explicar "por quê" adotaram uma certa abordagem.
- Dificuldades na verificação. Os SEs são difíceis de analisar o funcionamento interno e manutenção, já que fora necessário utilizar o conhecimento especialista na sua elaboração, esse conhecimento será necessário em etapas de manutenção.
- Pouca aprendizagem por experiência. Os SEs atuais são artesanais; quando o sistema já se encontra desenvolvido, o seu desempenho não irá melhorar sem a ajuda de seus programadores.

Silva (1987 apud Hoffman, 2008) afirma que o processo de aquisição do conhecimento para o SE caracteriza-se como o grande gargalo nos projetos de se desenvolver sistemas especialistas.

2.3 Raciocínio Baseado em Casos

A utilização de RBC consiste em buscar uma solução semelhante para um problema atual, através de uma experiência passada, armazenada na memória de casos; adaptar, se necessário, a solução ao problema atua; e armazenar o resultado na base de casos.

Essa metodologia evoluiu a partir do raciocínio baseado em regras, que possui certa similaridade com os sistemas especialistas. A grande diferença está na possibilidade do sistema reutilizar os resultados dos novos casos, para auxiliar na resolução de problemas similares no futuro.

2.3.1 Estrutura do RBC

De acordo com Riesbeck et al. (1989), um sistema de RBC, resolve os seus problemas, utilizando análises e adaptações que foram já utilizadas para resolver problemas antigos.

Segundo Wangenhein (2003), conhecimento, em um sistema de RBC, é principalmente armazenado na forma de casos, sendo caso uma peça de conhecimento contextualizado representando uma experiência passada ou episódio concretos. Contém a lição passada, que é o conteúdo do caso e o contexto em que a lição pode ser usada.

2.3.1.1 Componentes básicos de um caso

Conforme Wangenhein (2003), um caso é composto tipicamente dos seguintes componentes principais:

- O problema que descreve o estado do mundo quando o caso ocorreu;
- A solução que postula a solução derivada para aquele problema. A solução pode também ser uma ação, um plano ou uma informação útil ao usuário.

Watson (2000) explica que existe uma falta de consenso entre a comunidade de RBC sobre quais informações exatamente devem constar em um caso. De qualquer maneira duas

medidas pragmáticas podem ser levadas em consideração para decidir quais informações devem ser representadas em casos: a funcionalidade da informação e facilidade de aquisição da informação.

2.3.1.2 Similaridade

Wangenhein (2003) expõe que uma das hipóteses básicas de sistemas de RBC é que problemas similares possuem soluções similares. Com base nessa hipótese, o critério *a posteriori* da "utilidade de soluções" passa a ser reduzido ao critério *a priori* "similaridade de descrições de problema". Esta forma de proceder é justificada pela premissa de que, em muitas aplicações, a similaridade de definições de problemas implica a aplicabilidade da solução de um sobre o outro.

De acordo com Richter (1998), a formalização da similaridade pode ser obtida de duas formas: relacional e funcional. No caso da abordagem relacional, é utilizado uma relação de quatro operandos $R(x,y,u,v)$ de tal forma que "x e y são ao menos tão similares quanto u e v são". Dessa forma, obtém-se a seguinte definição da notação de vizinho mais próximo:

$$\mathrm{NN(x,z)} \Leftrightarrow (\forall_y)\, R(x,z,x,y)$$

Onde z é o vizinho mais próximo de x. Sendo único o vizinho mais próximo, então NN é também usado como símbolo de função. Na abordagem relacional, similaridade é tratada como ordenamento parcial. Tais ordenamentos parciais podem ser realizados por funções numéricas chamadas "medidas de similaridade" $sim(a,b)$ ou "medidas de distância" ou $d(a,b)$. Ambas medidas de similaridade *sim* e medidas de distância d induzem relações de quatro operandos R_{sim} e R_d de maneira lógica. Se $R_{sim}(x,y,u,v) \Leftrightarrow R_d(x,y,u,v)$ for verdadeiro, *sim* e d são ditos compatíveis.

Não obstante, Watson (1997) explica que apesar do aumento de complexidade, todos os algoritmos de vizinho mais próximo funcionam de forma similar. A similaridade (a proximidade) do caso alvo para um caso fonte para cada atributo é determinada. Essa medida

pode ser multiplicada por um fator de balanceamento. Então a soma da similaridade de todos os atributos é calculada. Isso pode ser representado pela relativamente simples equação :

$$Similaridade(T,S) = \sum_{i=1}^{n} f(T_i, S_i) \times w_i$$

Onde:

T é o caso alvo;

S é o caso fonte;

n é o número de atributos de cada caso;

i é um atributo individual de 1 para *n*

f é uma função de similaridade para atributo *i* em casos *T* e *S*

w é a importância balanceada do atributo *i*

2.3.2 Ciclo do RBC

Watson (1997) descreve o funcionamento do RBC como um processo cíclico que compreende 4 *R*s:

- Recuperar os casos mais similares.
- Reutilizar o caso na tentativa de resolver o novo problema.
- Revisar a solução proposta se necessário.
- Reter a nova solução como parte de um novo caso.

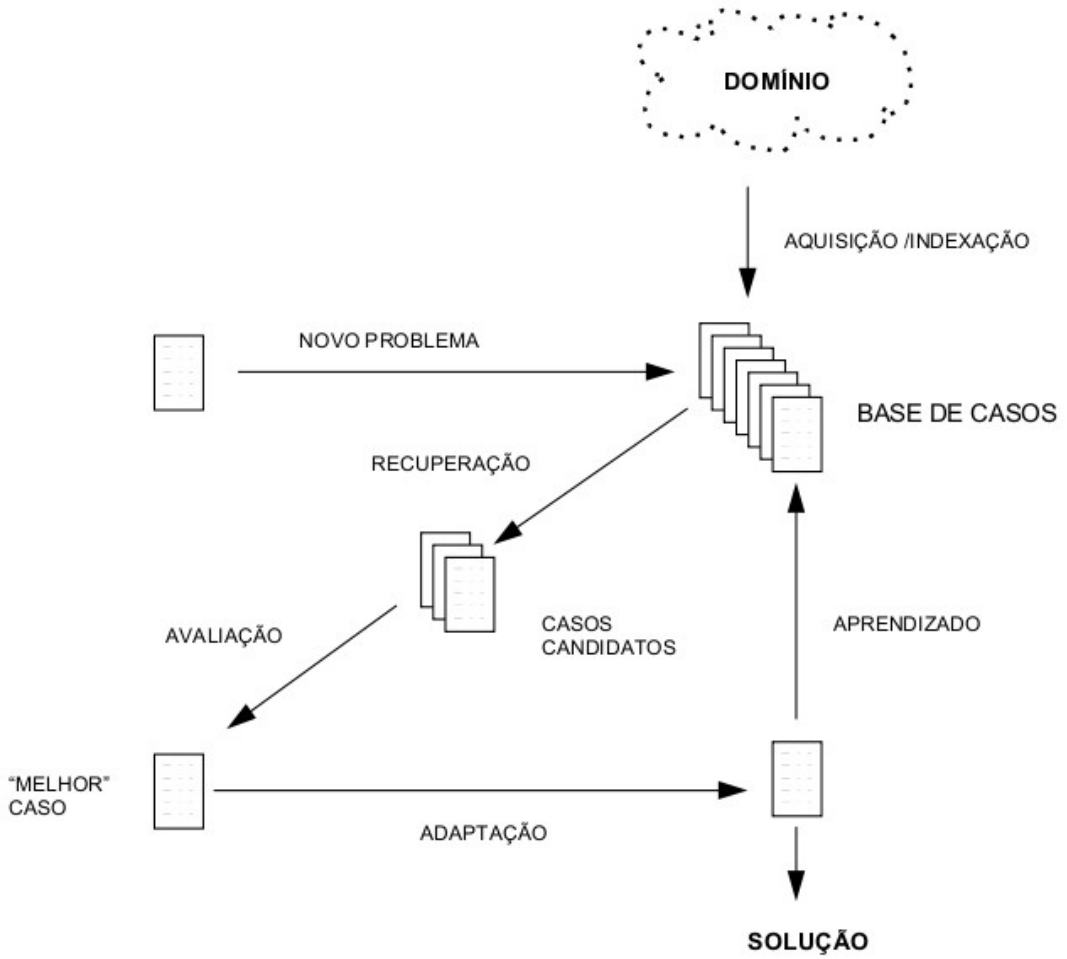

Figura 2.1 – Ciclo de solução de problema de um sistema CBR.
Fonte: (ABEL, 1996)

Utilizando-se dessa metodologia, o sistema de RBC tende acumular conhecimento na medida em que novos casos são solucionados e arquivados.

2.3.2.1 Recuperação de casos

De acordo com Richter (1998), a recuperação de dados é um processo comum em um banco de dados e, portanto em casos também. Uma consulta à um banco de dados retorna alguns dados através de uma exata combinação de chave-valor. A questão de combinação eficiente é chamada de problema da indexação. Uma consulta à um sistema de RBC presencia um problema e recupera uma solução usando combinações inexatas com os problemas armazenados na base de casos.

Segundo Wangenhein (2003) o objetivo da recuperação de casos é encontrar um caso ou um pequeno conjunto de casos na base de casos que contenha uma solução útil para o problema ou situação atual.

Para realizar essa recuperação, é necessário casar a descrição do problema atual com os problemas armazenados na base de casos, aplicando uma medida de similaridade.

2.3.2.1.1 Processo de Recuperação de Casos

Wangenhein (2003) define que o processo de recuperação de casos pode ser obtido através de um conjunto de subtarefas:

- Assessoramento da situação
- Casamento
- Seleção

No assessoramento da situação é formulada uma consulta descrevendo a situação ou problemas atuais, sendo a tarefa mais complexa das três, pois utiliza conhecimento e, muitas vezes exige uma interação pró-ativa com o usuário.

Problemas no mundo real, no entanto, podem possuir muitas características especiais, que por sua vez podem não se enquadrar na mesma forma que um problema armazenado na base de casos.

Dessa forma, o objetivo da recuperação de casos é obter da base de casos os casos potencialmente úteis na resolução do problema presente, através da similaridade entre o caso atual e os casos arquivados. Conforme Wangenhein (2003), isso é realizado através de uma consulta à base de casos utilizando como chave, descritores de entrada índices que são relevantes para encontrar a solução adequada. Esses descritores devem ser elaborados como a entrada do processo de recuperação.

Segundo Wangenhein (2003) a definição de quais descritores de entrada serão importantes depende do domínio de aplicação específico do sistema de RBC. A identificação

de quais descritores são relevantes para a solução do problema atual é um dos maiores fatores de complexidade de um sistema baseado em RBC.

Do casamento de dados pode resultar um conjunto de casos da base de casos útil para a solução do problema. Utilizam-se os descritores de entrada direta ou indireta para encontrar um conjunto de casos. Deve-se atenção ao fato de que em RBC o objetivo não é encontrar apenas os casos que casam completamente com o caso fonte, mas também casos potenciais, que casam parcialmente com os critérios de busca.

Wangenhein (2003) afirma que a partir do conjunto de casos similares, uma escolha considerada a melhor é realizada. Se um casamento demonstra ser forte o suficiente, uma tentativa de se encontrar um casamento melhor pode ainda ser realizada, por exemplo, com a exploração das diferenças com os outros casos do conjunto retornado. Se esta tentativa falha, tem-se uma evidência adicional da adequação do caso escolhido.

2.3.2.2 Reutilização de casos

Uma vez que um caso adequado é recuperado (WATSON, 1997), um sistema de RBC tentará reutilizar a solução sugerida pelo caso encontrado. Em muitas circunstâncias a solução poderá ser suficiente. Contudo, em outras circunstâncias a solução do caso recuperado poderá estar próxima a solução requerida, mas não próxima o bastante. O sistema de RBC deve então adaptar a solução armazenada no caso recuperado às necessidades do caso atual. A adaptação procura por diferenças proeminentes entre o caso recuperado e o caso atual e então aplica formulas ou regras que levam essas diferenças em consideração quando sugere uma solução final.

Em função disso, uma estratégia freqüentemente utilizada para construir sistemas de RBC é contornar o problema da adaptação inteiramente (LEA, 1996, apud WANGENHEIN, 2003), o que pode ser perfeitamente adequado para tarefas analíticas como classificação, diagnóstico ou suporte à decisão. Em vez de tentar adaptar casos recuperados à nova situação, a estratégia mais simples é a de inflar a base de casos ao máximo, para garantir que todo problema possível possua na base um caso cuja solução seja suficientemente similar, de maneira a não necessitar ser adaptada. Aplicações desse tipo muitas vezes utilizam-se de uma

filosofia de interação com o usuário em que o sistema sugere uma solução ou conjunto de soluções potenciais e um conjunto de ferramentas para navegação por esta solução, de maneira a permitir ao usuário explorar as respostas sob todos os aspectos e reutilizar a mais adequada às suas necessidades, modificando manualmente uma solução, se considerar necessário.

Para tarefas sintéticas, envolvendo configuração, projeto e planejamento de soluções, a natureza combinatória do espaço de solução invalida a estratégia de força bruta da inflação da base de casos. Segundo Wangenhein (2003), a adaptação inteligente de soluções encontradas torna-se essencial, em um sistema de RBC precisa ser capaz de adaptar a solução armazenada no caso encontrado na base de casos às necessidades da situação corrente.

Watson (1997) define várias técnicas, variando de simples a complexas, usadas para adaptação de casos, entre elas:

- Adaptação nula;
- Ajuste de parâmetros;
- Reinstanciação;
- Revisão derivacional;
- Reparo orientado a modelo.

No caso de adaptação nula, a solução encontrada é reutilizada no problema atual sem sofrer adaptações. Isso é útil para problemas envolvendo raciocínio complexo mas com uma solução simples.

O ajuste de parâmetros é uma técnica de adaptação estrutural que compara parâmetros especificados dos casos recuperado e atual para modificar a solução numa direção apropriada.

A reinstanciação é utilizada para instanciar características de uma solução antiga com novas características. Wangenhein (2003) propõe que reinstanciação é mais fácil de ser aplicada quando a estrutura geral de um problema anterior e sua solução são conhecidas.

A revisão derivacional consiste no processo de rastrear o método usado para chegar numa solução antiga (ou parte dela) para derivar uma solução na nova situação.

O reparo orientado a modelo usa um modelo causal para guiar a adaptação. Essa técnica também requer um bom entendimento do domínio do problema.

Watson (1997) explica que mesmo adaptação sendo útil em muitas situações, isso não significa que ela seja essencial e que a maioria dos sistemas de RBC comercializados não efetua qualquer adaptação. Eles simplesmente reutilizam a solução do caso com melhor casamento, ou deixam a adaptação para as pessoas.

2.3.2.3 Revisão de casos

Segundo Wangenhein (2003), quando uma solução para um caso gerada na fase de reuso não é correta, surge uma oportunidade para o aprendizado a partir desta falha.

De acordo com Silva (1997 apud Aamodt e Plaza, 2008) a revisão consiste de duas tarefas:

- Avaliação da solução gerada para reuso;
- Reparação da solução, utilizando o conhecimento especifico.

Wangenhein (2003) explica que o primeiro passo da revisão tem como prioridade a detecção de falhas nas soluções fornecidas pelo sistema de RBC. Depois que uma solução está completa, ou após certa parcela dela ter sido completada, a avaliação deve ser realizada.

A tarefa de avaliação pode tomar o resultado da aplicação da solução no ambiente de aplicação real por meio de monitoração automática resultados, ou pela interação com o usuário.

O eventual reparo de um caso envolve a detecção de quais partes da solução proposta contêm falhas e a recuperação ou geração de explicações para estas falhas conclui Wangenhein (2003). Quando um sistema inteligente necessita explicar uma falha, ele em

geral já resolveu o problema, tentou aplicar a solução ao mundo real e obteve algum tipo de *feedback* contando-lhe a respeito da falha. Alternativamente, o sistema pode ter simulado a aplicação da solução e ter chegado à conclusão de que a conclusão de que a solução não gera os resultados esperados.

2.3.2.4 Retenção de novos casos

Silva (2008) afirma que um sistema em RBC, somente será considerado eficiente quando estiver preparado por aprender devido a sua base de experiências já absorvidas e possuindo uma correta indexação.

Wangenhein (2003) propõe que a retenção de casos é o processo de incorporação, ao conhecimento já existente, daquilo que é útil de um novo episódio de solução de um problema. O objetivo de se reter continuamente conhecimento toda vez que um novo problema é resolvido é o de constantemente atualizar e estender a base de casos. Isto permite a um sistema de RBC continuamente incrementar seu conhecimento e tornar-se um solucionador de problemas mais poderoso, com o passar do tempo e sua utilização.

Ainda segundo Wangenhein (2003), a retenção automática de novo conhecimento em RBC, quando for além do simples armazenamento de um caso cuja solução foi confirmada como certeza, utiliza técnicas de “aprendizado de máquina”.

Richter (1998) explica que os métodos de aprendizado de máquina podem ser utilizados na melhoria da base de conhecimento de um sistema de RBC, em particular: na base de casos, executando tarefas de adição, criação e remoção de casos; na medida de similaridade, ajustando balanceamentos; e na transformação da solução, aprendendo novas regras de adaptação.

Silva (2008) destaca que o aprendizado do sistema de RBC acontece, quase que sempre, em sua maior parte, como um subproduto do raciocínio, armazenando novas experiências na memória de casos e, disponibilizando-as para futuros casos.

De acordo com Wangenhein (2003), podemos distinguir três tipos básicos de retenção em sistemas de RBC:

- Sem retenção;
- Retenção de soluções de problemas;
- Retenção de documentos.

Sendo que quando não há retenção de casos, significa que o sistema de RBC não armazenará novos casos e estes deverão ser criados e armazenados em tempo de desenvolvimento. Essa abordagem está presente em sistemas de RBC mais simples, sendo aplicado principalmente em domínios de aplicação bem compreendidos.

Já na retenção de soluções de problemas, as novas experiências são retidas na medida em que novos problemas são resolvidos, de forma a auxiliar na solução de problemas similares no futuro. Assim sendo, sempre que um novo problema é resolvido, este pode ser incorporado à base de casos como caso novo.

Wangenhein (2003) acrescenta que alguns sistemas de RBC também retêm casos representando tentativas de solução de problemas que falharam, para permitir que se evite utilizar aquelas soluções no futuro.

Na retenção de documentos, novo conhecimento é adquirido de forma assíncrona ao processo de solução de problemas. Assim, o conhecimento é adquirido de forma independente da operação do sistema, sempre que se encontrar disponível, como por exemplo, documentos em um sistema de gerência do conhecimento (GLW, 2001, apud WANGENHEIN, 2003).

A retenção equivale a aprendizado de novos casos ou de conhecimentos relevantes abstraídos de casos individuais. Aprendizado efetivo requer um conjunto de métodos bem trabalhado, de forma a extrair conhecimento relevante da experiência passada, indexar este conhecimento para uso posterior e integrar casos em uma estrutura existente.

Os principais aspectos a serem considerados durante a fase de retenção de casos são:

- Seleção adequada da informação a ser armazenada conjuntamente com o caso;
- Seleção da estrutura da informação e do conhecimento;
- Seleção da estrutura de índices para o acesso aos casos durante a recuperação, e;
- Seleção do tipo de integração a ser realizado na estrutura de conhecimento existente.

Wangenhein (2003) afirma que o "problema da indexação" é um problema central em RBC. Ele implica decidir que tipo de índices devem ser utilizados para recuperações futuras e como estruturar o espaço de busca. Isto é, na verdade, um problema de aquisição de conhecimento e deveria ser analisado como parte do passo inicial de análise do domínio e modelagem de conhecimento. Uma solução trivial para esse problema com certeza é a utilização de todas as entidades de informação como índices.

O passo final da retenção de novos casos é a integração do novo conhecimento na base de conhecimento. Isto inclui a adição de novos casos na base de casos ou a modificação ou remoção de casos antigos com utilização de enfoques de aprendizado.

2.3.3 Vantagens do RBC

Conforme Watson (1997) demonstra, a característica do RBC de combinações inexatas de casos é uma grande vantagem sobre sistemas típicos de recuperação de informações. Tais sistemas baseiam–se em combinações exatas de dados, o que pode ocultar informações próximas às pesquisadas, como por exemplo sinônimos.

RBC tem se mostrado mais preciso no diagnóstico de defeitos de produtos como demonstra Watson (1993 apud Nakhaeizadeih, 1997) na comparação entre técnicas de RBC e análise de discriminante linear, efetuada na Daimeler-Benz, no controle de qualidade de caixas de câmbio automotivas. Eles tinham mais de 7000 casos, cada um deles estava descritos com mais de 57 atributos de características.

Tabela 2.1 – Porcentagem da precisão de RBC versus análise de discriminante linear

Teste	RBC	Análise de discriminante linear
1	93,4%	61%
2	93,4%	60%
3	92,6%	62%
4	93,2%	61%
5	93,5%	62%
Média	93,2%	61%

Fonte: (NAKHAEIZADEIH, 1997, apud WANGENHEIN, 2003)

A tabela 2.1 sumariza os resultados dos experimentos da Daimler-Benz, mostrando que RBC foi em torno de 30% mais preciso em classificar diagnósticos do que o método estatístico.

Watson (1997) demonstra que sistemas de RBC diferem profundamente de sistemas especialistas de forma que para usar RBC não devemos saber como resolver um problema, mas apenas reconhecer se já resolvemos um problema similar no passado. Existindo, dessa forma, uma redução no esforço de captação de conhecimento, já que sistemas de RBC não requerem o entendimento de "como" resolver o problema.

Wangenhein (2003) destaca que a retenção de casos propicia uma das maiores vantagens específicas de um sistema de RBC, pois o aprendizado e a aplicação do conhecimento aprendido não são separados de forma estrita, mas sim integrados.

Watson (1997) acrescenta que a habilidade de adquirir novos casos durante o uso sem ter que adicionar novas regras ao sistema, nem modificar as existentes.

Segundo Watson (1997) redes neurais podem ser menos eficientes que RBC em domínios que envolvam dados complexos, e simbolicamente estruturados. Mas a maior vantagem de RBC sobre redes neurais recai na característica de que redes neurais funcionam como "caixas pretas". A resposta obtida de uma rede neural é uma função dos vetores balanceados de seus neurônios. Nem explicações ou justificativas de qualquer tipo podem ser obtidas por uma rede neural. Portanto, nesse aspecto, redes neurais são ainda piores que sistemas baseados em regras ou sistemas de aprendizado de máquina, nos quais ao menos as regras utilizadas podem ser utilizadas para justificar uma decisão.

Abaixo, seguem algumas outras vantagens para com a utilização de RBC, segundo os pontos de vista que merecem evidência (PALM, 2004, apud SILVA, 2008).

- Redução da tarefa de aquisição de conhecimento;
- Evita repetir enganos feitos no passado;
- Fornece flexibilidade dentro da modelagem de conhecimento;
- Argumenta domínios que não foram completamente compreendidos, definidos ou modelados.
- Faz predições do sucesso provável de uma solução oferecida;
- Aprende com o passar do tempo;
- Evita repetir todos os passos que precisam ser levados para chegar a uma solução.

Enfim, diante de todas as vantagens relacionadas à utilização de RBC, não há duvidas de que a reflexão do raciocino humano é o que mais se destaca (SILVA, 2008).

3 PROBLEMATIZAÇÃO DA PROPOSTA

Este capítulo apresentará o problema a ser resolvido com o emprego de RBC, bem como o detalhamento de contribuições e objetivos a serem alcançados com este trabalho.

3.1 Identificação do problema de pesquisa

Uma das mais importantes funcionalidades dos SGRT é a gerência de falhas. A gerência de falhas é fortemente baseada nos registros obtidos a partir do monitoramento de equipamentos. Assim, pode-se dizer que para alcançar um bom funcionamento, um SGRT deve extrair informação a partir dos *logs* obtidos a partir do monitoramento dos equipamentos gerenciados.

Segundo Cisco (2006) a mais básica e, ao mesmo tempo, a tarefa mais importante consiste de simplesmente recolher os eventos da rede e garantir que nada de importante se perdeu. Isso inclui a recepção de eventos e armazená-los na memória, para que possam continuar a ser processados por um aplicativo ou um operador humano que decide como reagir. A funcionalidade de gerenciamento de base de eventos também inclui a persistência de eventos, ou seja, escrevê-las em disco ou armazená-los em um banco de dados, para construir um registro histórico de eventos que ocorreram.

De acordo com Farrel et al. (2009) uma das atribuições da gerência de falhas é prover mecanismos de filtragem de eventos de falha ou alarmes e correlação de eventos para a redução do número de eventos relevantes e para a análise da causa raiz.

Cisco (2006) enfatiza que dados históricos de eventos podem ser utilizados para ajudar com diagnóstico de futuro e de correlação. Basicamente, isso pode ser útil para identificar padrões de alarme que ocorreram de forma semelhante em tempos passados. Reconhecendo tais padrões, e recuperando a sua resolução pode ajudar a resolver problemas futuros mais rápido. Podem ser usados para estabelecer as tendências, para ver como as taxas de alarme e os tipos de alarmes evoluíram ao longo do tempo. Eles podem ser analisados em conjunto com outros dados históricos, tais como as alterações que foram realizadas na rede, por exemplo, a introdução de novos elementos da rede e seu impacto sobre os padrões históricos de alarmes, ou correlação de alarmes com determinados padrões de uso da rede.

A maneira mais simples de analisar o resultado da monitoração de equipamentos gerenciados consiste em exibir ao usuário as notificações, enviadas pelos equipamentos aos SGRT, ou mudanças ocorridas nos equipamentos percebidas pelos SGRT. Nesse caso, para obter informações, os usuários de SGRT podem lançar mão de ferramentas de filtragem e agregação de registros.

Segundo Kay (2003), quanto mais complexa uma rede for e quanto mais as aplicações são distribuídas, mais eventos, alarmes e alertas aos sistemas de monitoramento irão gerar no caso de uma única falha.

Kay (2003) cita o exemplo de Chris Jordan, um gerente de segurança em Computer Sciences Corp., o qual relatou que conexões OC-12 podem gerar 850 MB de dados de eventos em uma hora. (OC-12 é uma conexão de fibra ótica com banda de 622 Mbit/s). O que se transforma em mais de 600 GB de dados por mês, ou 7TB por ano – Isto apenas para *logs* e alertas relacionados a um único *link*.

Verifica-se que a quantidade de *log* gerado por uma planta de equipamentos de telecomunicações pode ser gigantesca, o que dificulta a obtenção de informação, indispensável para a correta tomada de decisão. Nesse cenário também é dificultado a identificação de tendências, e análise pró ativa na resolução de problemas.

De acordo com os problemas identificados, elaborou-se uma proposta que possa elucidar as seguintes questões:

- Que tipo de técnica poderia efetuar análise de registros de *log* de SGRT;

- Que metodologia deveria ser empregada para possibilitar que as causas de problemas possam ser encontradas.

- Que ferramenta poderia fazer o SGRT "aprender" com os problemas já resolvidos pelos usuários.

Ao pesquisar as ferramentas computacionais disponíveis, encontra-se no abrangente campo da inteligência artificial os mecanismos necessários para resolver o problema em questão.

Segundo Krishnamoorthy (1996) a tecnologia de inteligência artificial fornece técnicas para o desenvolvimento de programas de computador para a realização de uma variedade de tarefas, simulando a maneira inteligente de seres humanos de resolver problemas.

Dentro da área de inteligência artificial, existem várias subdivisões de técnicas em constante pesquisa e aprimoramento. Dentre as técnicas pesquisadas e com bom nível de amadurecimento, com qualidades que interessam na resolução do problema encontram-se: Sistemas Especialistas, Sistemas Multiagentes, Redes Neurais e Raciocínio Baseado em Casos.

De acordo com Luger (2004) um sistema especialista usa conhecimento específico de um domínio de problema para conseguir um desempenho com "qualidade de perícia" naquela área de aplicação. De um modo geral, os projetistas de sistemas especialistas adquirem esse conhecimento com a ajuda de peritos humanos no domínio, e o sistema emula a metodologia e a atuação do perito humano. Os sistemas especialistas, assim como as pessoas experientes tendem a ser especialistas, focando sobre um conjunto reduzido de problemas.

Um sistema multiagente, como define Luger (2004), é um programa de computador com solucionadores de problemas localizados em ambientes interativos, que são capazes de ações flexíveis, autônomas e, ainda socialmente organizadas que podem, mas não necessariamente, ser dirigidas para objetivos predeterminados ou metas. Assim, os quatro critérios para um sistema de agentes inteligentes incluem solucionador de problemas que são localizados, autônomos, flexíveis e sociais.

Russel (2004) define redes neurais como funções não-lineares complexas com muitos parâmetros. Seus parâmetros podem ser aprendidos a partir de dados ruidosos, e elas podem ser usadas em milhares de aplicações.

O RBC, como explica Wangenheim (2003), tem suas raízes inspiradas nos trabalhos de Schank e Abelson sobre Memória Dinâmica e no modelo cognitivo de uma lembrança de situações passadas (casos e memória episódica) e de padrões de situações. A Memória Dinâmica tem origem na tentativa de se criar modelos cognitivos da solução de problemas e do aprendizado de situações com base em memórias episódicas.

Se analisarmos a dinâmica da análise de eventos de SGRT, verificamos o diagnóstico de problemas na maioria das vezes é a partir da repetição de situações que seguem determinado padrão.

RBC exige pouco esforço do desenvolvedor no aprimoramento da base de conhecimento utilizado para inferir resultados, já que essa tarefa acontece naturalmente no ciclo do RBC. Ao passo que em sistemas especialistas é necessário que a equipe de desenvolvimento de software implemente as regras. Em redes neurais, para novas inferências é necessário recalcular a rede neural. E em sistemas multiagentes é necessário alterar o comportamento dos agentes inteligentes, o que demanda investimentos na produção de software durante o ciclo de vida da utilização do sistema.

Pode-se afirmar que todas as técnicas de inteligência artificial listadas, e não exclusivamente elas, poderiam ser utilizadas na correlação de eventos de SGRT, obtendo

resultados muito satisfatórios. Porém em sistemas de apoio operacional, como no caso o SGRT, a reduzida carga em manutenção para utilização do sistema é muito desejada.

Portanto, foi escolhido a ferramenta de RBC para atuar na melhoria da obtenção de informações a partir de dados de eventos colhidos por SGRT.

3.2 Contribuição dessa pesquisa e projeto

Ao verificar-se os dados que são obtidos por um SGRT, constata-se que os registros apontam o estado de diferentes mecanismos presentes nos equipamentos gerenciados.

Um acontecimento típico é a geração de *logs* semelhantes quando as repetidas ações são realizadas nos equipamentos gerenciados.

Essa é a característica que deverá possibilitar a correlação de eventos, baseando-se em situações semelhantes, já ocorridas no passado. Uma vez que o SGRT já possuir os registros conseqüentes de ações e/ou problemas já ocorridos, na eventual recorrência de tais eventos o sistema poderá apontar as correções necessárias que já foram utilizadas nos casos anteriores.

Outra característica importante do uso de RBC na correlação de eventos consiste na possibilidade de alimentar a base de casos com novos casos, pois muitas situações podem não ser previstas quando no desenvolvimento de um SGRT.

3.3 Proposta de modelagem de sistema baseado em RBC

Segundo OMG (2005), a modelagem é a concepção de aplicações de software antes da codificação. Usando um modelo, os responsáveis por um projeto de desenvolvimento de software podem assegurar-se que as funcionalidade são completas e corretas, as necessidades do usuário final sejam cumpridos, e o projeto do sistema suporta os requisitos de escalabilidade, robustez, segurança, extensibilidade, e outras características, antes da execução.

Este trabalho pretende apresentar a modelagem de um sistema que, utilizando RBC, possa extrair informações úteis a partir de *logs* de eventos de SGRT. Na elaboração dessa modelagem, pretende-se verificar a viabilidade técnica da ferramenta de RBC, bem como sua capacidade de adaptação ao contexto.

As principais atribuições do sistema modelado serão a obtenção de informações obtidas a partir da correlação de eventos obtidos de SGRT reais. Para tanto, serão utilizados como base o modelo de dados de um SGRT. Isso leva a pressupor que a coleta de dados e guarda dos *logs* será fornecida pelo SGRT.

O SGRT deverá fornecer os *logs* ao sistema modelado, e este será encarregado de encontrar correlação entre os registros fornecidos. Dessa forma o sistema proposto deverá guardar o histórico de cados obtidos, bem como os dados que forem necessários.

O sistema de correlação proposto armazenará casos que corresponderão a um conjunto de eventos, juntamente com uma descrição do caso e uma solução possível, quando aplicável. Para recuperar um caso o sistema varrerá um conjunto de eventos comparando seus atributos com os atributos dos casos armazenados. Deverão ser utilizados os critérios de similaridade definidos na modelagem para decidir se um caso é cadidato ou não para ser utilizado.

A modelagem do sistema de correlação poderá servir como base para implementação de um sistema, a ser utilizado em conjunto com um SGRT.

As informações obtidas da correlação de eventos possibilitarão maior eficiência no diagnóstico de problemas e acontecimentos na rede de equipamentos monitorada, reduzindo o tempo necessário na análise de *logs*.

4 MODELAGEM DE SISTEMA DE CORRELAÇÃO DE EVENTOS

4.1 Introdução

A modelagem do Sistema de Correlação visa apresentar o projeto de um software capaz de efetuar a correlação de eventos consolidados por um SGRT.

Wieringa (2003) afirma que a Unified Modeling Language (UML) é uma coleção de técnicas de diagramação adotado pelo Object Management Group (OMG) em 1997 como notação padrão para projetos de software orientado a objetos. A UML foi criada para trazer um fim à enorme diversidade de notação utilizada em diferentes métodos orientados a objetos.

De acordo com Yeates et al. (2004) métodos Orientados a Objeto (OO) permitem que o desenvolvedor explore a tecnologia de ambientes de computação distribuída, sistemas baseados na Internet e software de comunicações e ferramentas. A abordagem de métodos estruturados, emergindo em uma era de desenvolvimento de sistemas corporativos e software de base de dados centralizada, muitas vezes era considerado inadequado para o desenvolvimento da rede, sistemas de interface gráfica, especialmente quando esses sistemas foram desenvolvidos de forma incremental usando a computação distribuída

Langer (2007) explica que a análise de sistemas utilizando orientação a objetos resulta em melhores sistemas que são mais coesos, reutilizáveis e de fácil manutenção. Mais coeso é código mais “curto”, e é mais facilmente corrigido e mantido. É também a base de componentes reutilizáveis que podem ser incorporados em outras aplicações mais tarde. Sem orientação a objetos, os sistemas tendem a possuir código fonte que são duplicadas e difícil manutenção.

Segundo Langer (2007), hoje, a análise orientada a objeto se tornou uma questão chave do paradigma de análise. Sendo sem dúvida o elemento mais importante de criar o que pode ser chamado de o "completo" requisito de um sistema. Infelizmente, a indústria está em controvérsia sobre as abordagens e ferramentas que devem ser utilizados para criar sistemas de objetos. A orientação a objeto é baseada no conceito de que todos os requisitos em última análise, devem pertencer a um objeto. Por isso, é fundamental primeiro definir o que se entende por um objeto. No contexto da análise OO, um objeto é qualquer conjunto coeso formado por dois componentes essenciais: dados e processos.

Na elaboração da modelagem é definido um escopo do que o software deverá atender, e a partir do escopo são especificados os casos de utilização, os utilizadores e a estrutura necessária.

4.2 Requisitos

Segundo Larman (2002), os requisitos são capacidades e condições que o sistema - e, mais amplamente, o projeto - devem conformar-se. Um primeiro desafio do trabalho de levantar requisitos é encontrar, comunicar e lembrar (que normalmente significa registro) o que é realmente necessário, de uma forma clara para o cliente e os membros da equipe de desenvolvimento.

Foram levantados requisitos a partir da literatura pesquisada, verificando-se as atribuições dos SGRT, os objetivos da análise de eventos e as ações necessárias para utilizar RBC na correlação de eventos.

4.2.1 Receber eventos

O sistema deverá obter os dados a partir de uma base de dados, efetuando queries sql nas tabelas utilizadas pelo SGRT.

4.2.2 Armazenar eventos

Os eventos deverão ser armazenados em base de dados.

Cada registro de evento deverá discriminar os seguintes dados:

- Data/hora com precisão de mínima de segundos. Preferencial precisão de milissegundos.
- Identificador do dispositivo a que se refere o evento. Pode ser um endereço IP, nome de placa.
- Modelo do equipamento.
- Severidade. Informa o quão grave é o evento. Podendo assumir um dos seguintes estados: Normal, Warning, Minor, Major ou Critical. (Normal, Aviso, Menor, Maior, Crítico)
- Descrição do evento.

4.2.3 Armazenar casos

Os casos deverão ser armazenados em base de dados para garantir disponibilidade e agilidade nas consultas.

4.2.4 Listar eventos

O sistema deverá apresentar uma listagem dos eventos (logs) recebidos do SGRT. A lista de eventos deverá permitir seleção de múltiplos registros.

4.2.5 Relatar caso

O usuário deverá poder relatar um caso, para tal, o usuário deverá selecionar um ou vários eventos que correspondem a um caso e descrever esse caso.

4.2.6 Correlacionar eventos

O usuário poderá selecionar um grupo de eventos por intervalo de tempo, equipamentos ou ambos, e aplicar a correlação a eles. Como resultado, deverão ser apresentados casos semelhantes, que constem na base de casos.

4.3 Atores

4.3.1 SGRT

O SGRT representa o Sistema de Gerenciamento de Redes de Telecomunicações. Abrange suas rotinas, interfaces de integração e sua base de dados.

É tido como responsável por configurar e monitorar a rede de equipamentos. Utilizando protocolos e dispondo de facilidades para consolidação de dados.

A atribuição mais importante do SGRT para a correlação de eventos consiste na obtenção dos eventos de equipamentos heterogêneos, conforme descrito na seção 1.3.2 Gerenciamento de Falhas, utilizando variados protocolos de comunicação, e consolidando os dados em entidades únicas, armazenadas em base de dados, o que possibilita consultas ágeis.

4.3.2 Usuário

O usuário do Sistema de Correlação será encarregado de selecionar eventos e informar os casos que estes refletem. Todo o suporte necessário ao ciclo do RBC ficará a cargo do usuário, na forma de ações para criação e manutenção de casos.

O usuário também operará a ferramenta na busca de informações a partir de eventos. Em condições reais de utilização da ferramenta modelada, esses papéis poderão ser exercidos por funcionários distintos numa empresa que mantém uma rede de equipamentos gerenciados.

4.4 Casos de uso

Larman (2002) explica que a UML provê a notação de diagramas de casos de uso para ilustrar os nomes dos casos de uso e atores, e os relacionamento entre eles. (...) Um simples diagrama de caso de uso provê um contexto visual para o sistema, ilustrando os atores externos e como eles usam o sistema.

Ainda de acordo com Larman (2002), um diagrama de casos de uso é uma excelente figura de contexto do sistema; ele faz um bom diagrama de contexto, ou seja, mostrando as fronteiras de um sistema, o que se encontra no exterior do sistema e como isso pode ser usado. Ele serve como uma ferramente de comunicação que sumariza o comportamento de um sistema e seus atores.

Yen et al. (1998) define que o propósito do diagrama de caso de isso é representar um diagrama de contexto. Entendendo o diagrama de caso de uso, pode-se entender os atores externos ao sistema, e as múltiplas formas pelas quais o sistema é utilizado pelos atores.

Os casos de uso identificam os momentos de utilização possíveis do Sistema de Correlação. A especificação de casos de uso permite obter uma visão geral das ações que podem ser efetuadas no sistema.

Todos os casos de uso identificados podem ser visualizados no diagrama de casos de uso apresentado abaixo:

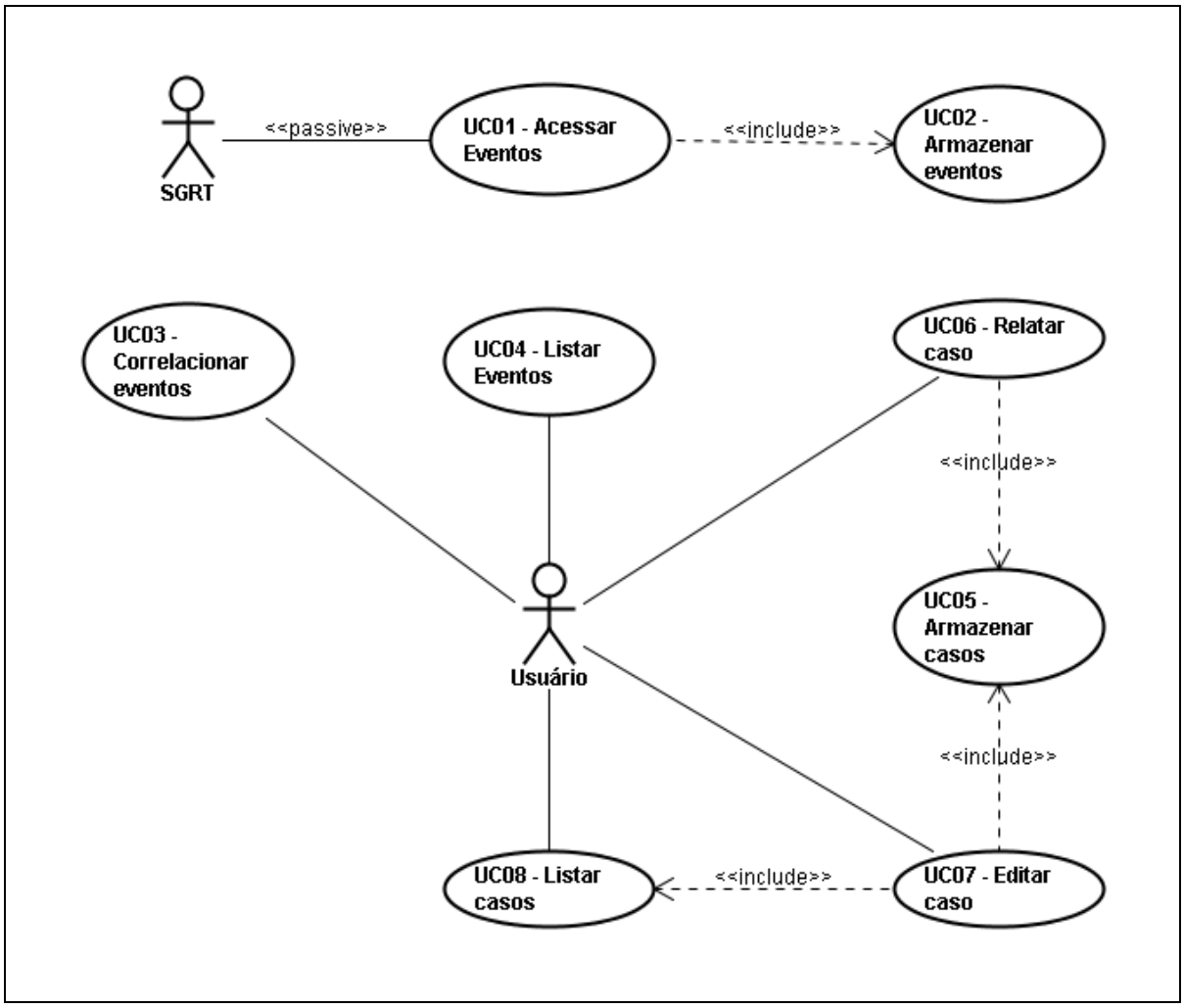

Figura 4.1 – Diagrama de Casos de Uso

Fonte: Autor

Cada caso de uso é descrito para obter-se a especificação de como o mesmo será realizado.

De acordo com Larman (2002) podem ser totalmente descritos. Casos de uso totalmente descritos exibem mais detalhes e são estruturados; eles são úteis para obter um entendimento profundo das metas, tarefas e requisitos.

Nas próximas seções deste trabalho é apresentada uma descrição detalhada de cada caso de uso identificado.

4.4.1 UC01 – Acessar Eventos

Para dar início a correlação de eventos, é necessário primeiramente obter os registros que serão correlacionados. Para tal tarefa, foi modelado o acesso direto à base de dados do SGRT. Os SGRT consolidam os dados de eventos ocorridos e os armazenam na sua base de dados. Essa característica pode ser utilizada para exportar dados para outros sistemas, como nesse caso, para um sistema de correlação.

Dessa forma, a integração é rápida e com bom desempenho, e garante um bom nível de robustez a ambos sistemas, já que o sistema de correlação não alterará dados do SGRT, e o sistema de correlação contará com dados mais atualizados possíveis.

Para fins de otimização e eficiência da ferramenta de correlação, a cada consulta a base de dados do SGRT, apenas dados ainda não recebidos pelo sistema deverão ser retornados pelas consultas.

Descrição do caso de uso:
Consiste na funcionalidade de acessar os registros de eventos do SGRT. É efetuado acesso à base de dados do SGRT.
Atores envolvidos:
SGRT
Fluxo básico (Happy Day):
1. São obtidos os dados de acesso a base de dados; 2. O sistema efetua consulta na base de dados do SGRT; 3. Registros ainda não persistidos são enviados à camada de persistência;

Fluxos Alternativos (Exceptions):
Falta de conectividade à base de dados do SGRT 1. Se ocorrer *timeout* na tentativa de acesso a base; 2. O sistema exibe mensagem informando o erro. **Falta de permissão para acesso à base de dados do SGRT** 1. Se retornar um erro de permissão negada na tentativa de acesso ao SGBD; 2. O sistema exibe mensagem informando o erro.
Regras de negócio:
Não se aplica.
Pré-condições:
Os dados oriundos da base de dados do SGRT deverão estar em conformidade com o modelo entidade-relacionamento implementado na ferramenta de correlação. Deve ser permitido acesso à base de dados do SGRT Deve ser permitido acesso às tabelas que armazenam dados referentes aos eventos gravados pelo SGRT.
Pós-condições:
Não se aplica.
Relacionamento com Outros Casos de Uso:
UC02 - Armazenar eventos

Quadro 4.1 – Descrição do Caso de uso UC01 - Acessar Eventos

Fonte: Autor

4.4.2 UC02 - Armazenar eventos

Os eventos obtidos do SGRT deverão ser armazenados em base de dados própria do sistema de correlação. A utilização de base de dados própria eleva o nível de independência e robustez do sistema de correlação, e diminui a carga no SGRT, o que é uma característica desejável.

Descrição do caso de uso:
O sistema deverá persistir registros de eventos em base de dados própria.
Atores envolvidos:
SGRT
Fluxo básico (Happy Day):
1.A camada de persistência recebe dados para inclusão. 2.O sistema verifica se os eventos já não existem na base de dados própria para evitar repetições, de acordo com regra de negócio RN01 - Unicidade de evento. 3.O sistema insere na base de dados própria os registros correspondentes.

Fluxos Alternativos (Exceptions):
Falta de conectividade à base de dados própria 1.Se ocorrer *timeout* na tentativa de acesso a base; 2.O sistema exibe mensagem informando o erro. **Falta de permissão para acesso à base de dados própria** 1.Se retornar um erro de permissão negada na tentativa de acesso ao SGBD; 2.O sistema exibe mensagem informando o erro.
Regras de negócio:
RN01 - Unicidade de evento
Pré-condições:
Dados de entrada deverão ser fornecidos.
Pós-condições:
Não se aplica.
Relacionamento com Outros Casos de Uso:
UC01 – Acessar Eventos

Quadro 4.2 – Descrição do Caso de uso UC02 - Armazenar Eventos

Fonte: Autor

4.4.3 UC03 – Correlacionar Eventos

A correlação de eventos é o principal componente do sistema de correlação. Ele acessará dos dados da sua base de dados própria, e, utilizando técnicas de busca baseando-se em similaridade, tentará encontrar casos já ocorridos, que tenham gerado seqüências de eventos semelhantes.

A maneira de calcular a similaridade entre diferentes casos pode ser verificada na seção 2.3.1.2 Similaridade. O quadro 4.1 demonstra um exemplo de corrrelação.

<table>
<tr><td>Caso relatado</td><td></td><td>Similaridade</td></tr>
<tr><td>Eventos</td><td>
<table>
<tr><td>Modelo</td><td>Severidade</td><td>Hora</td><td>Descrição</td></tr>
<tr><td>Modem x</td><td>Crítical</td><td>00:00:00</td><td>Erro no envio de dados</td></tr>
<tr><td>Modem x</td><td>Major</td><td>00:00:03</td><td>Erro CRC</td></tr>
<tr><td>Modem x</td><td>Warning</td><td>00:00:06</td><td>Modem reiniciado</td></tr>
</table>
</td><td></td></tr>
<tr><td>Modelo de equipamento</td><td>Modem x</td><td></td></tr>
<tr><td>Duração</td><td>6 segundos</td><td></td></tr>
<tr><td>Número de eventos</td><td>3</td><td></td></tr>
<tr><td>Severidades</td><td>1 x Critical
1 x Major
1 x Warning</td><td></td></tr>
<tr><td></td><td></td><td></td></tr>
<tr><td>Caso Candidato 1</td><td></td><td></td></tr>
<tr><td>Descrição</td><td>Cabo rompido</td><td></td></tr>
<tr><td>Eventos</td><td>
<table>
<tr><td>Modelo</td><td>Severidade</td><td>Hora</td><td>Descrição</td></tr>
<tr><td>Modem x</td><td>Crítical</td><td>00:00:00</td><td>Falha na conexão</td></tr>
<tr><td>Modem x</td><td>Crítical</td><td>00:00:02</td><td>Erro no envio de dados</td></tr>
<tr><td>Modem x</td><td>Major</td><td>00:00:03</td><td>Falta de sincronismo</td></tr>
<tr><td>Modem x</td><td>Major</td><td>00:00:05</td><td>Erro CRC</td></tr>
<tr><td>Modem x</td><td>Major</td><td>00:00:05</td><td>Timeout</td></tr>
</table>
</td><td>30%</td></tr>
<tr><td>Modelo de</td><td>Modem x</td><td>100%</td></tr>
</table>

<table>
<tr><td>equipamento</td><td></td><td></td></tr>
<tr><td>Duração</td><td>5 segundos</td><td>83%</td></tr>
<tr><td>Número de eventos</td><td>5</td><td>60%</td></tr>
<tr><td>Severidades</td><td>2 x Critical
3 x Major</td><td>30%</td></tr>
<tr><td>Pontuação</td><td>30 positivo
5 negativo
Saldo = 20
Total = 35</td><td></td></tr>
<tr><td></td><td></td><td></td></tr>
<tr><td>Caso Candidato 2</td><td></td><td></td></tr>
<tr><td>Descrição</td><td>Blindagem do cabo defeituosa
Distância de conexão muito longa</td><td></td></tr>
<tr><td>Eventos</td><td><table><tr><td>Modelo</td><td>Severidade</td><td>Hora</td><td>Descrição</td></tr><tr><td>Modem x</td><td>Crítical</td><td>00:00:00</td><td>Erro no envio de dados</td></tr><tr><td>Modem x</td><td>Major</td><td>00:00:03</td><td>Erro CRC</td></tr></table></td><td>66%</td></tr>
<tr><td>Modelo de equipamento</td><td>Modem x</td><td>100%</td></tr>
<tr><td>Duração</td><td>3 segundos</td><td>50%</td></tr>
<tr><td>Número de eventos</td><td>2</td><td>66%</td></tr>
</table>

Severidades	1 x Critical 1 x Major	66%
Pontuação	45 positivo 2 negativo Saldo = 43 Total = 47	

Quadro 4.3 – Exemplo de Correlação de Eventos

Fonte: Autor

Para podermos comparar os casos ilustrados, devemos analisar os índices ponderando seus valores de acordo com os critérios de similaridade definidos na RN02 – Critérios de similaridade:

Tabela 4.1 – Aplicação de critérios de similaridade

Critério	Peso	Caso candidato 1	Caso candidato 2
Duração do caso	10%	83% x 0,1 = 8,3	50% x 0,1 = 5
Modelos de equipamento	20%	100% x 0,2 = 20	100% x 0,2 = 20
Descrições dos eventos associados	25%	30% x 0,25 = 7,5	66% x 0,25 = 16,5
Número de eventos associados	15%	60% x 0,15 = 9	66% x 0,15 = 9,9
Severidades dos eventos associados	10%	30% x 0,1 = 3	66% x 0,1 = 6,6
Saldo de pontuação positiva	10%	20 x 0,1 = 2	43 x 0,1 = 4,3
Total de avaliações do caso	10%	35 x 0,1 = 3,5	47 x 0,1 = 4,7
Pontuação alcançada		53,3	67

Fonte: Autor

Percebe-se que os casos candidatos apresentados no Quadro 4.3 eram muito semelhantes, mas os diferentes eventos associados a cada caso levaram a pontuações distintas em relação ao caso relatado. No exemplo exposto, o Caso Candidato 2 tem maior semelhança com o Caso Relatado.

Descrição do caso de uso:
Consiste em, a partir de um conjunto de registros de eventos, efetuar uma consulta na base de casos cadastrados utilizando critérios de similaridade.
Atores envolvidos:
Usuário
Fluxo básico (Happy Day):
1.O usuário seleciona um conjunto de eventos que ele considere relevante. 2.O sistema percorre os eventos selecionados, efetuando consultas a base de casos usando como critério de similaridade a regra de negócio RN02 - Critério de similaridade. 3.Os casos encontrados são exibidos ao usuário e este pode avaliar se o caso faz sentido.

Fluxos Alternativos (Exceptions):
Nenhum evento selecionado 1.Se nenhum evento for selecionado, o sistema não tem como efetuar correlação. 2.Uma mensagem de aviso é enviada ao usuário. **Nenhum caso encontrado** Se a busca por casos não retornar dados, não foi possível estabelecer correlação entre os eventos selecionados. Isso pode ocorrer por dois motivos: número de eventos selecionados insuficiente ou casos relevantes ainda não cadastrados. Uma mensagem de aviso é enviada ao usuário.
Regras de negócio:
RN02 - Critério de similaridade
Pré-condições:
Um ou mais eventos deverão ter sido selecionados para efetuar-se correlação. Deverão existir casos já cadastrados na base de casos.
Pós-condições:
Não se aplica.
Relacionamento com Outros Casos de Uso:
UC04 - Listar Eventos UC05 - Armazenar Casos

Quadro 4.4 – Descrição do Caso de uso UC03 - Correlacionar Eventos

Fonte: Autor

4.4.4 UC04 - Listar Eventos

A listagem de eventos exibirá os eventos já extraídos dos SGRT e armazenados em base de dados própria do sistema de correlação. A lista de eventos possibilitará restringir os grupos de eventos nos quais se tentará obter correlação.

Descrição do caso de uso:
Consiste em listar os registros de eventos obtidos do SGRT. Essa lista deverá contar com filtros de acordo com os atributos dos eventos. A lista deverá possibilitar a seleção de múltiplos eventos.
Atores envolvidos:
Usuário.
Fluxo básico (Happy Day):
1.O usuário opcionalmente define um filtro. 2.É efetuada uma consulta na base de dados própria em busca de eventos obtidos do SGRT. 3.Os eventos são listados para o usuário. 4.O usuário poderá selecionar um ou vários eventos listados. Dessa forma, habilitam-se as opções de relatar caso e correlacionar eventos.

Fluxos Alternativos (Exceptions):
Base de eventos vazia 1.Se não existirem eventos na base de dados própria. 2.Uma mensagem de aviso deverá ser exibida ao usuário, informando que pode ser necessário importar registros de eventos do SGRT. **Nenhum evento encontrado** 1.Se os critérios de filtro selecionados não corresponderem a qualquer evento. 2.Uma mensagem de aviso deverá ser exibida ao usuário, informando que pode ser necessário alterar algum critério na busca.
Regras de negócio:
Não se aplica.
Pré-condições:
Eventos já devem ter sido importados da base de dados do SGRT.
Pós-condições:
Não se aplica.
Relacionamento com Outros Casos de Uso:
UC03 - Correlacionar Eventos UC06 - Relatar Caso

Quadro 4.5 – Descrição do Caso de uso UC04 - Listar Eventos

Fonte: Autor

4.4.5 UC05 - Armazenar Casos

Os casos já identificados pelo sistema deverão ser registrados em base de dados para facilitar a pesquisa de casos no futuro. Como o sistema de correlação baseia-se em casos passados para desempenhar suas funções, a pesquisa na base de casos deverá ser o mais eficiente possível.

Quando na utilização do sistema de correlação, num primeiro momento, os usuários deverão criar casos que reflitam situações reais, cada caso com seu conjunto de eventos. Após ter-se um bom número de casos registrados é que o sistema de correlação poderá ser utilizado com eficiência.

Descrição do caso de uso:
Consiste em armazenar em base de dados própria o caso relatado.
Atores envolvidos:
Usuário.
Fluxo básico (Happy Day):
1.Os dados referentes ao caso são recebidos, juntamente com um conjunto de eventos que foram selecionados pelo usuário. 2.O sistema sumariza os dados do caso, de acordo com a regra de negócio RN03 – consolidar caso. 3.Os dados são persistidos na base de dados, alimentando a base de casos do sistema de correlação.

Fluxos Alternativos (Exceptions):
Falta de conectividade à base de dados própria 1.Se ocorrer *timeout* na tentativa de acesso a base; 2.O sistema exibe mensagem informando o erro. **Falta de permissão para acesso à base de dados própria** 1.Se retornar um erro de permissão negada na tentativa de acesso ao SGBD; 2.O sistema exibe mensagem informando o erro.
Regras de negócio:
RN03 – Consolidar Caso
Pré-condições:
Base de dados própria deve estar acessível.
Pós-condições:
Não se aplica.
Relacionamento com Outros Casos de Uso:
UC03 - Correlacionar Eventos UC06 - Relatar Caso

Quadro 4.6 – Descrição do Caso de uso UC05 - Armazenar Casos

Fonte: Autor

4.4.6 UC06 - Relatar Caso

Ao relatar um caso, o usuário estará alimentando a base de casos. A partir da identificação de uma seqüência de eventos, o usuário poderá designar que esse conjunto de

eventos corresponde a algum comportamento no sistema, seja algum problema, marco ou apenas algum aviso.

Posteriormente, caso busque-se correlação numa seqüência de eventos semelhante, o caso relatado pelo usuário deverá ser exibido.

Descrição do caso de uso:
Consiste em receber um conjunto de eventos selecionados pelo usuário e adicionar dados relevantes que caracterizarão o caso.
Atores envolvidos:
Usuário.
Fluxo básico (Happy Day):
1.São recebidos eventos selecionados pelo usuário. 2.Usuário descreve o caso, que reflete os eventos selecionados. 3.Sistema valida caso de acordo com RN04 – Validação Caso. 4.O caso é enviado para base de casos.
Fluxos Alternativos (Exceptions):
Validação do caso 1.Se usuário não descrever todos os campos obrigatórios descritos na RN04 – Validação Caso. 2.Uma mensagem de erro é exibida ao usuário.
Regras de negócio:
RN04 – Validação Caso

Pré-condições:
Deverão ser recebidos eventos que farão parte da descrição do caso.
Pós-condições:
Não se aplica.
Relacionamento com Outros Casos de Uso:
UC04 - Listar Eventos UC05 - Armazenar Casos

Quadro 4.7 – Descrição do Caso de uso UC06 - Relatar Caso

Fonte: Autor

4.4.7 UC07 - Listar Casos

Pode ser necessário efetuar manutenções na base de casos para o bom funcionamento da ferramenta de correlação. Para tanto, deverá ser implementadas funcionalidades para isso.

Wangenheim (2003) explica que uma solução que foi adaptada é validada e, se necessário corrigida. Esse passo representa uma parte essencial do processo de aprendizado humano: a confirmação de uma solução plausível para um problema. Critérios para esta revisão da solução podem ser a correção e qualidade da solução, bem como critérios específicos da aplicação em questão, como a facilidade de compreensão da solução por parte do usuário ou a quantidade de esforço para implementar-se a sugestão sugerida. A verificação da solução é, em muitos casos, realizada durante sua aplicação na prática, embora possa ser substituída por uma simulação.

Uma forma de realizar a revisão dos casos poderá ser pela lista de casos, em que o usuário pesquisa os casos pelos seus atributos, e então realiza modificações para enriquecer a base de casos com soluções ou melhorias nas descrições de casos e descrições de soluções para problemas encontrados.

Descrição do caso de uso:
Essa ferramenta deverá ter caráter secundário, agindo na manutenção da base de casos. Com ela o usuário poderá remover casos que julgar errados, ou com baixo índice de acerto. Consiste em listar os casos já relatados pelo sistema de correlação. Deverá possibilitar a seleção de casos para possível manutenção como edição ou exclusão do caso. Deverá possuir ferramenta de filtro, de acordo com os atributos presentes nos casos.
Atores envolvidos:
Usuário
Fluxo básico (Happy Day):
1.Usuário seleciona um filtro adequado. 2.Sistema busca da base de casos os casos que correspondem ao filtro informado. 3.São exibidos os casos encontrados. 4.O usuário poderá selecionar casos para exclusão ou abrir tela de edição para alterar características de algum caso.

Fluxos Alternativos (Exceptions):
Base de casos vazia 1.Se não existirem registros de casos na base de dados. 2.Uma mensagem de aviso deverá ser exibida ao usuário. **Nenhum caso encontrado** 1.Se os critérios de filtro selecionados não corresponderem a qualquer caso. 2.Uma mensagem de aviso deverá ser exibida ao usuário, informando que pode ser necessário alterar algum critério na busca.
Regras de negócio:
Não se aplica.
Pré-condições:
Casos já devem ter sido cadastrados anteriormente.
Pós-condições:
Não se aplica.
Relacionamento com Outros Casos de Uso:
UC08 - Editar caso

Quadro 4.8 – Descrição do Caso de uso UC07 - Listar Casos

Fonte: Autor

4.4.8 UC08 - Editar Caso

Dentro da etapa do RBC de revisão de casos, é fornecida uma forma de editar as propriedades de qualquer caso armazenado pelo sistema.

Descrição do caso de uso:
Consiste em exibir uma tela para edição de algum caso já cadastrado na base de casos. Deverá ser utilizada para efetuar alterações pontuais nos casos, tais como corrigir erros de digitação por exemplo. O usuário deverá ser advertido a fazer alterações com atenção, pois isso poderá comprometer a eficiência do sistema de correlação de eventos.
Atores envolvidos:
Usuário.
Fluxo básico (Happy Day):
1.Usuário abre a tela de edição de eventos. 2.Os dados do caso são populados na tela. 3.O usuário altera os dados que julgar necessário. 4.O usuário manda salvar alterações. 5.O sistema valida o caso de acordo com a regra de negócio RN04 – Validação Caso.
Fluxos Alternativos (Exceptions):
Validação do caso 1.Se usuário não descrever todos os campos obrigatórios descritos na RN04 – Validação Caso. 2.Uma mensagem de erro é exibida ao usuário.
Regras de negócio:
RN04 – Validação Caso

Pré-condições:
Um caso deve ser informado, ou selecionado da lista de casos.
Pós-condições:
Não se aplica.
Relacionamento com Outros Casos de Uso:
UC07 - Listar casos.

Quadro 4.9 – Descrição do Caso de uso UC08 - Editar Caso

Fonte: Autor

4.5 Regras de Negócio

Segundo Langer (2007), as regras de negócio são a lógica da aplicação que impõem a integridade do negócio, ou seja, que mantém as regras como definidas pelos próprios usuários.

As regras de negócio especificam as diversas restrições e validações presentes no sistema. Seu detalhamento em separado possibilita aprofundar a análise sobre partes importantes do projeto.

4.5.1 RN01 – Unicidade de evento

Para que a correlação de eventos seja eficiente, a base de dados do sistema de correlação deverá conter registros únicos para cada evento obtido do SGRT. Essa condição deverá ser observada na sucessiva obtenção de dados do SGRT.

Descrição:
Os eventos armazenados pelo sistema deverão ser únicos, para se alcançar a unicidade de evento, deverá ser adotada a seguinte abordagem: Chave primária: o identificador único do evento no SGRT não deverá ser propagado diretamente para o sistema de correlação, em vez disso, ele será armazenado numa coluna da entidade evento. Dessa forma, ao consultar os eventos do SGRT deverão ser obtidos apenas os que contenham chave primária diferente dos eventos armazenados no sistema de correlação. No sistema de correlação, os eventos deverão possuir chave primária independente da chave primária dos eventos no SGRT.
Casos de Uso afetados:
UC02 – Armazenar eventos

Quadro 4.10 – Regra de negócio RN01 - Unicidade de Evento

Fonte: Autor

4.5.2 RN02 - Critério de Similaridade

O critério de similaridade é peça chave na realização da correlação de eventos. A partir da definição dos critérios de similaridade é possível correlacionar eventos semelhantes ocorridos em ocasiões diferentes.

Wangenheim (2003) afirma que os índices de um caso são combinações de seus atributos mais importantes, que permitem distingui-lo de outros e identificar casos úteis para uma dada descrição de problema.

Descrição:
Os critérios de similaridades utilizados na correlação dos eventos deverão ser os seguintes:

•Duração do caso. Peso 10 %. •Modelos de equipamentos afetados. Peso 20 %. •Descrições dos eventos associados. Peso 25 %. •Número de eventos associados. Peso 15 %. •Severidades dos eventos associados. Peso 10 %. •Saldo de pontuação positiva. Peso 10 %. •Total de avaliações do caso. Peso 10 %.
Casos de Uso afetados:
UC03 - Correlacionar eventos

Quadro 4.11 – Regra de negócio RN02 - Critério de Similaridade

Fonte: Autor

4.5.3 RN03 – Consolidar Caso

Na consolidação de casos, são obtidos dados a partir do relato de caso. Dessa forma, os dados relevantes ao caso são armazenados juntamente com o mesmo.

Essa consolidação ajudará o mecanismo de correlação a encontrar casos com atributos semelhantes aos eventos alvos da correlação.

Descrição:
Deverão ser obtidos a partir da lista de eventos do caso: • O tempo total de duração dos casos. • Os modelos de equipamentos envolvidos.

Casos de Uso afetados:
UC05 - Armazenar casos

Quadro 4.12 – Regra de negócio RN03 - Consolidar Caso

Fonte: Autor

4.5.4 RN04 – Validação Caso

Na validação de casos são verificados os atributos do caso para verificar ser os dados mínimos para definir com clareza um caso estão presentes.

Descrição:
Para considerar que um caso seja válido, os seguintes dados deverão ser informados: • Data inicial do caso. • Data final do caso. • Descrição do caso. • Modelos de equipamentos envolvidos. • Lista de eventos associados ao caso.
Casos de Uso afetados:
UC06 - Relatar Caso UC08 - Editar Caso

Quadro 4.13 – Regra de negócio RN04 - Validação Caso

Fonte: Autor

4.6 Diagrama de classes

Yen et al. (1998) esclarece que diagramas de classe especificam as classes do *software* e interfaces numa aplicação. Em geral eles incluem classes, atributos, suas operações

ou métodos, e as associações entre as classes. Nota-se que o diagrama de classes representa as entidades de software em vez de entidades do modelo conceitual.

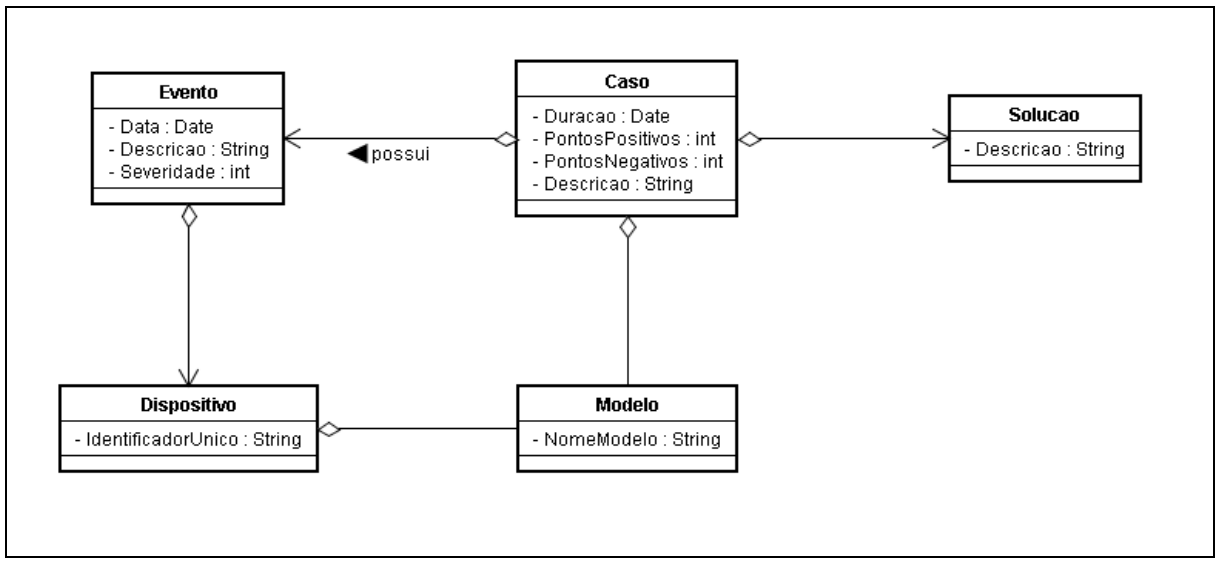

Figura 4.2 – Diagrama de classes

Fonte: Autor

4.7 Diagramas de seqüência

De acordo com Yen et al. (1998), diagramas de seqüência são um tipo de diagrama de interação desenhado usando a notação UML, que retrata a interação entre objetos e mostra detalhadamente o fluxo de mensagens entre objetos dentro de um caso de uso; a linha do tempo é direcionada para baixo e os objetos são representados nas colunas verticais.

Nas próximas páginas serão apresentados diagramas que descrevem em alto nível os principais fluxos de processamento do sistema de correlação de eventos.

Dos diagramas são apresentadas interações das entidades Sistema de Correlação e DAO, além dos atores Usuário e SGRT.

A entidate DAO representa a camada da ferramenta de correlação responsável pelo acesso às bases de dados do SGRT bem como da própria base de dados do sistema de correlação.

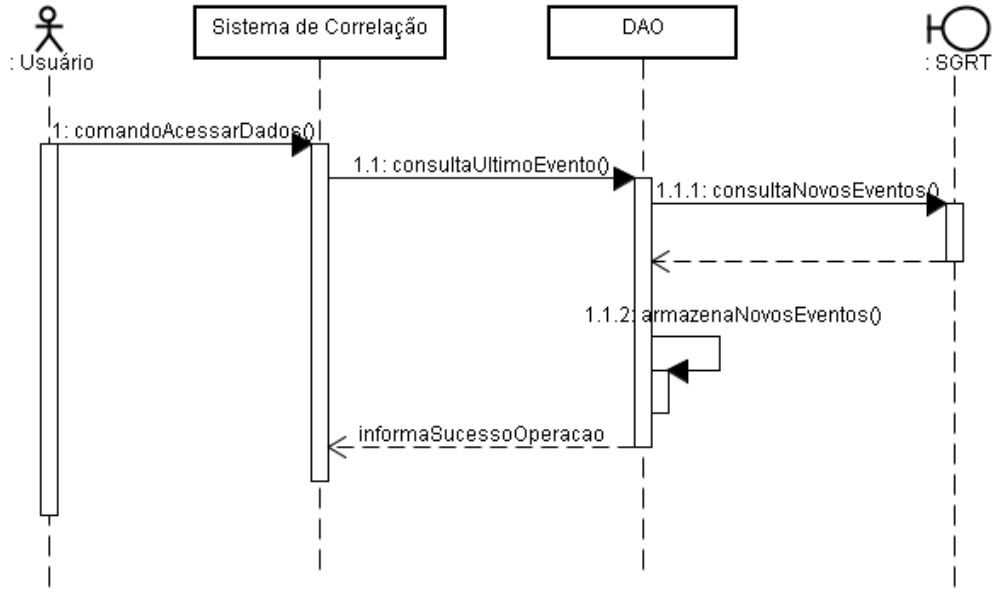

Figura 4.3 – Diagrama de seqüência da integração com SGRT

Fonte: Autor

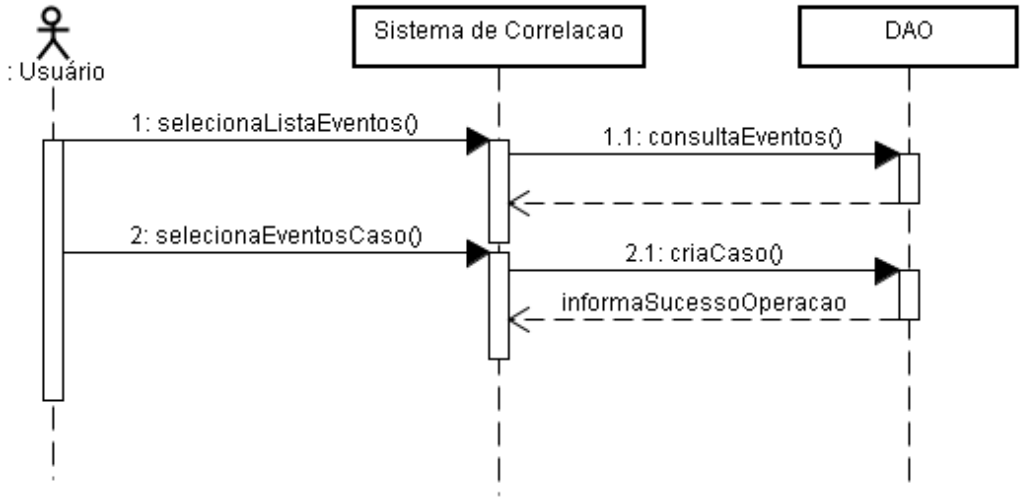

Figura 4.4 – Diagrama de seqüência do relato de novos casos

Fonte: Autor

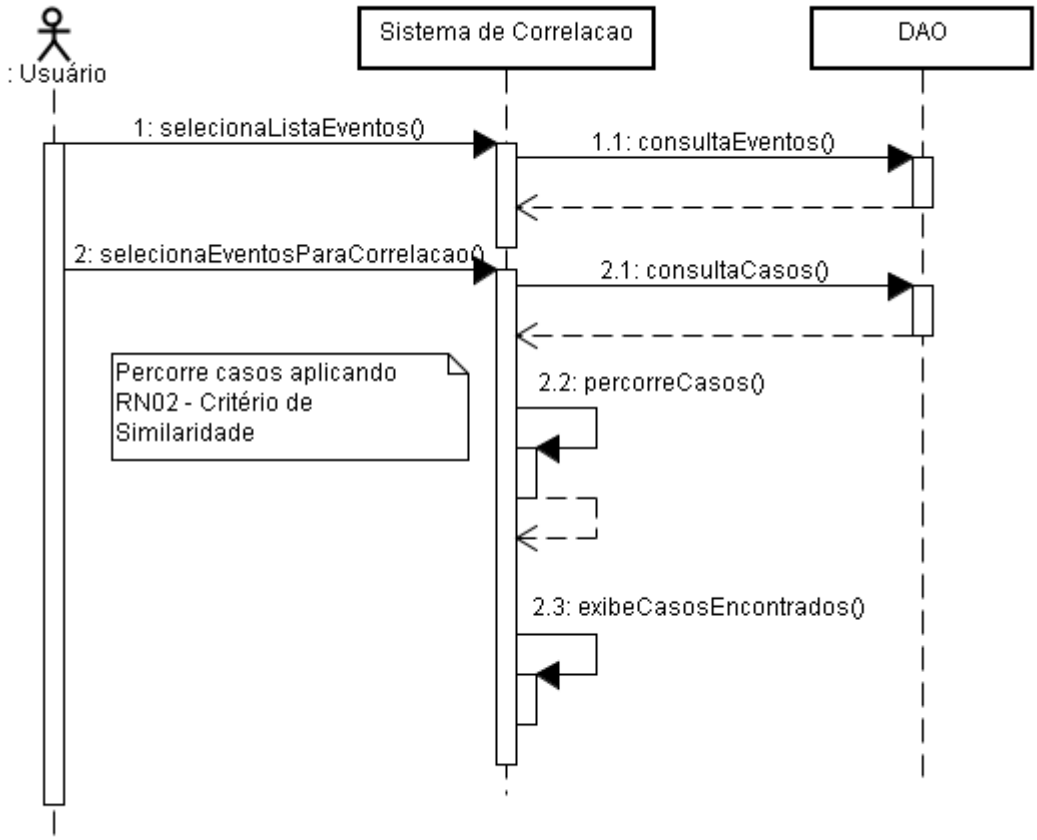

Figura 4.5 – Diagrama de seqüência da correlação de eventos

Fonte: Autor

4.8 Protótipos de telas

Segundo Blaha (2006), uma interface com o usuário é o objeto ou um conjunto de objetos que fornece ao usuário de um sistema uma forma de acessar seus objetos, comandos e opções de aplicação.

A seguir serão apresentados protótipos de telas que podem seguir de base para a implementação do sistema. Dessa forma, é possível visualizar a forma que a ferramenta poderá se tornar.

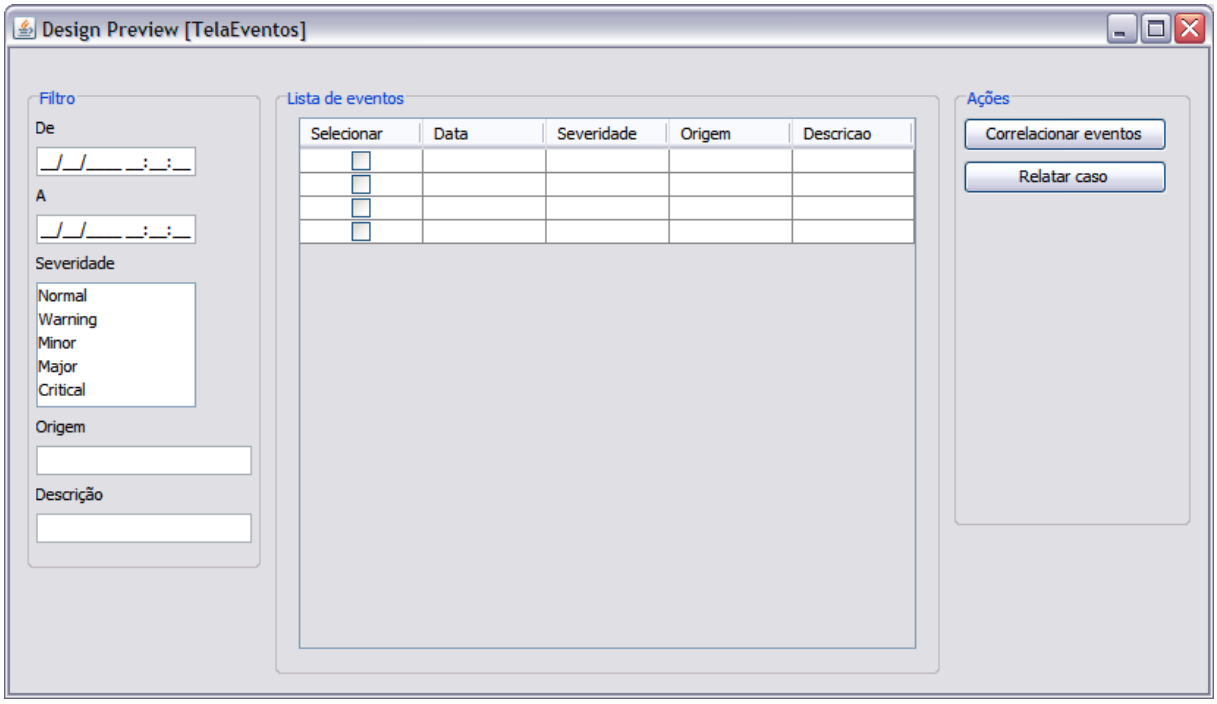

Figura 4.6 – Protótipo de tela de lista de eventos

Fonte: Autor

A figura 4.6 exibe a tela de listagem de eventos, a qual conta com ferramentas de filtro e seleção bem como ações para relatar caso e correlacionar eventos.

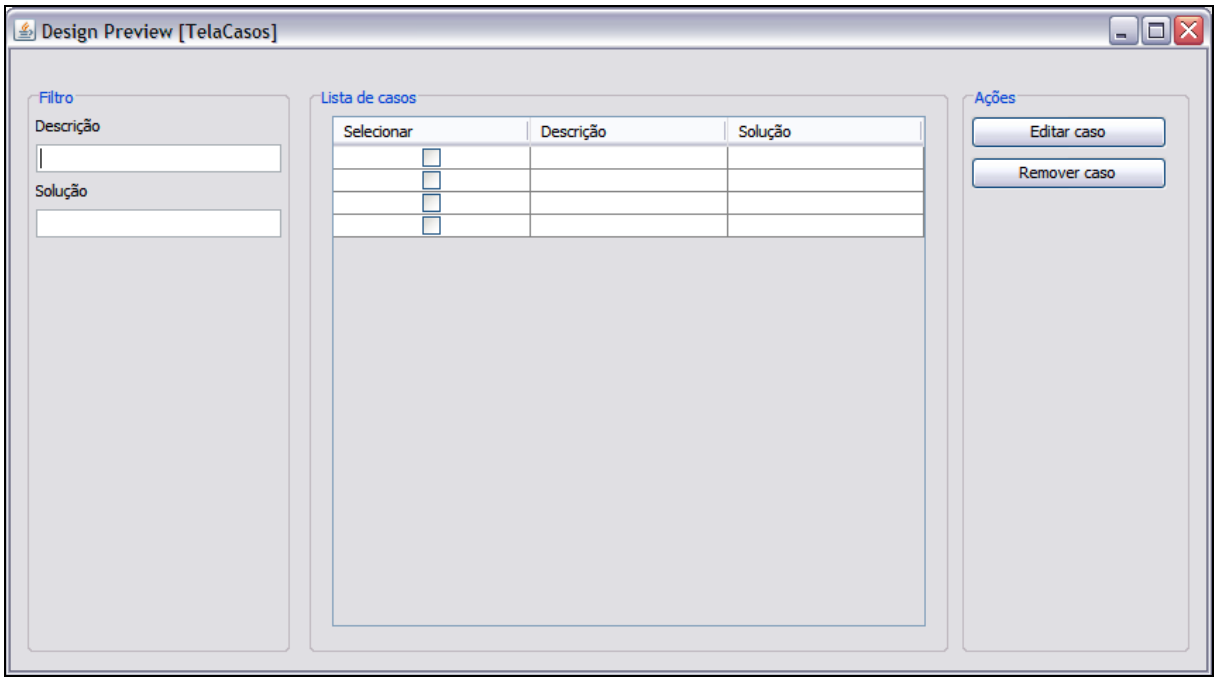

Figura 4.7 – Protótipo de tela de lista de casos

Fonte: Autor

Na figua 4.7 é exibida a tela para listar casos. Essa tela possui ferramentas de filtro de casos, e ações para efetuar manutenção nos casos listados.

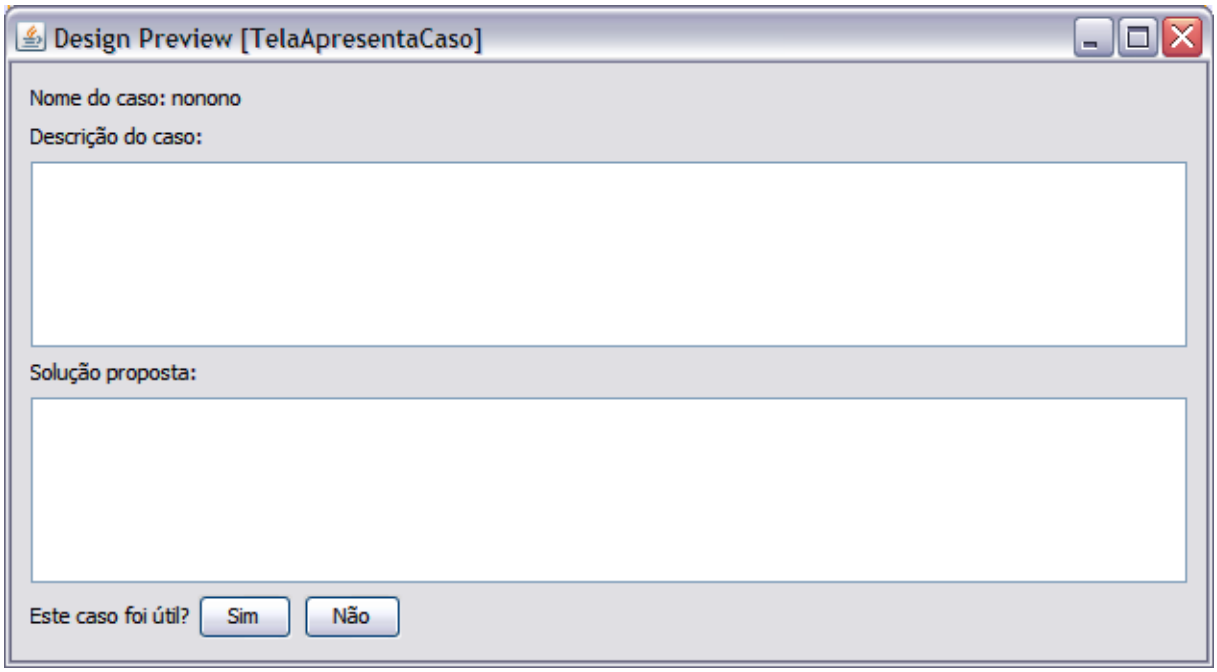

Figura 4.8 – Protótipo de tela de apresentação de caso

Fonte: Autor

Na figura 4.8 é exibido a tela de apresentação de caso, esta tela deverá mostrar os detalhes de um caso que corresponda a uma correlação de eventos.

Nessa tela é possível obter informações a respeito do caso, bem como adicionar informações como melhorar a descrição ou inserir uma solução para um eventual problema.

A tela conta ainda com ações simples de pontuação, que podem ajudar na busca pelos melhores casos. Na medida em que o caso for reutilizado, mais pontos ele poderá acumular.

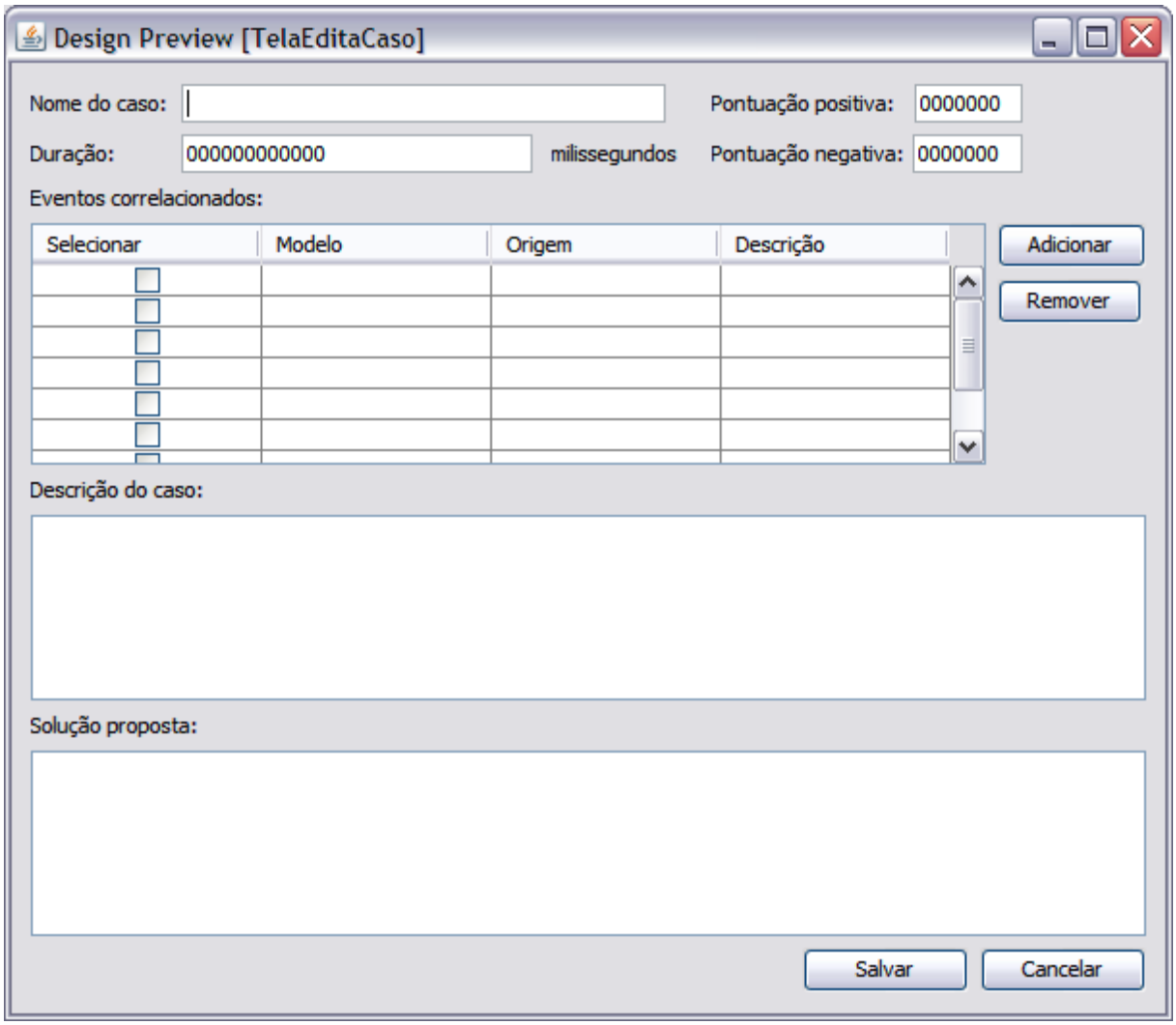

Figura 4.9 – Protótipo de tela de edição de caso

Fonte: Autor

Na Figura 4.9 é exibida a tela de edição de caso. Essa tela serve para manutenção avançada de casos. Caso algum caso cadastrado tenha erros que estejam prejudicando a correlação de eventos, este pode ser reparado nessa tela, ou até mesmo removido permanentemente na tela de listagem de casos.

CONCLUSÃO

Os SGRT têm assumido uma grande importância, na medida em que as evoluções nas redes de telecomunicações modificaram e incrementaram as redes já existentes. Dessa forma, as tarefas que os SGRT desempenham aumentaram em elevados graus de complexidade.

Nesse contexto, os equipamentos têm adotado protocolos de gerenciamento simples que enviam dados pouco processados aos SGRT.

Um exemplo de dado pouco processado submetido aos SGRT são os *logs* de eventos ocorridos nos equipamentos, utilizados para sinalizar o funcionamento da rede de telecomunicações.

Para diagnosticar falhas ocorridas nos equipamentos, pode-se recorrer aos *logs* de eventos armazenados no SGRT. Porém a análise isolada por usuários desses registros demonstra-se custosa e de resultados limitados no diagnóstico de problemas na rede.

Para auxiliar no diagnóstico, pode ser utilizada alguma técnica de inteligência artificial. Automatizando a tarefa e inspecionando a fundo os problemas que ocorrem na rede, com grande agilidade se comparado ao trabalho manual executado por mãos humanas.

Existem várias técnicas de inteligência artificial disponíveis, e nem todas possuem a flexibilidade de se adaptar a novos padrões de problemas.

A metodologia de RBC pode ser utilizada com grandes benefícios, pois tem relativa facilidade de implementação, capacidade de aprendizagem e flexibilidade de se adaptar entre problemas diversos.

A modelagem de sistema possibilita obter um projeto de software bem definido e documentado, aumentando a qualidade do software projetado. A partir da modelagem pode-se avaliar com mais clareza aspectos de mais alto nível de um software, tais como arquitetura e pontos de integração. É possível distinguir melhor os pontos mais importantes do software, permitindo adotar medidas adequadas em tempo de projeto.

Durante a modelagem do sistema, foi possível verificar detalhes de funcionamento do sistema que interferiram na sua arquitetura como um todo. Caso essas características fossem descobertas durante a implementação do sistema, o trabalho de manutenção seria expressivo. Esse foi um exemplo claro da facilidade adquirida no desenvolvimento ao adotar etapas de análise e modelagem de software.

Este trabalho limitou-se a análise e projeto de um sistema de correlação. As definições em alto nível apresentadas aqui poderão ser utilizadas na implementação de um sistema em uso junto aos SGRT existentes ou mesmo incluído no projeto de SGRT com poucas adaptações.

Um ponto de extensão possível deste trabalho é a utilização de mais dados de SGRT para formar uma base de casos com mais atributos de correlação, como por exemplo dados sobre os estados de configuração de cada equipamento.

A ferramenta de RBC se mostrou muito flexível, foi possível observar potencial de aplicação em outros problemas da área de gerenciamento de redes de telecomunicações, e sistemas de apoio operacional em geral.

REFERÊNCIAS BIBLIOGRÁFICAS

ABEL, M.; CASTILHO, J. M.V. **Um estudo sobre Raciocínio Baseado em Casos**, PPGC-UFRGS, 1996.

BERNSTEIN, Lawrence; YUHAS, C. M. **Basic Concepts for Managing Telecommunications Networks**: Copper to Sand to Glass to Air. ed Springer, 1999.

BLAHA, Michael; RUMBAUGH, James. **Modelagem e projetos baseados em objetos com UML 2.** 2. ed., rev. atual. Rio de Janeiro: Elsevier, 2006

BURKHARD, Hans-Dieter. **Extending some Concepts of Case Based Reasoning – Foundations of Case Retrieval Nets**. In:Lenz, M. Bartsch-Spörl, B. Burkhard H-D. Wess, S. (Eds.): Case-Based Reasoning Technology from foundations to Applications, Springer-Verlag, Heidelberg, Germany, 1998, p.17-50.

CARVALHO, Flavia; JUNIOR, Aguinaldo F.; SILVEIRA, Jorge G.; LUNARDELLI, Fernando; AZAMBUJA, Marcelo. **FreeNMS** – Uma Plataforma Software Livre para Gerência e Administração de Nível de Serviço (SLM). Laboratório MetroPoA (PUCRS), Porto Alegre, 2002. Disponível em <http://www.rnp.br/_arquivo/wrnp2/2003/oscfa05a.pdf>. Acessado em 15 jun 2009.

CARVALHO, Flavia; JUNIOR, Aguinaldo F.; SILVEIRA, Jorge G.; LUNARDELLI, Fernando. **Sistemas TTS**: Uma abordagem voltada para Sistemas de Gerenciamento de

Redes. Laboratório MetroPoA (PUCRS), Porto Alegre, 2003. Disponível em <http://www.rnp.br/_arquivo/wrnp2/2003/oscfa04a.pdf>. Acessado em 15 jun 2009.

CISCO. **Cisco Transport Manager User's Guide, 9.0**. Disponível em: <http://www.cisco.com/en/US/docs/net_mgmt/cisco_transport_manager/9.0/user/guide/usergd90.html>. Acesso em: 30 jun. 2009.

CISCO. **Network Management Fundamentals:** A guide to understand how network management technology really works. United States: Cisco Press. 2006.

CYCLADES. **Guia Internet De Conectividade**. ed. Senac. 6a. ed. São Paulo, 2000.

FARREL, Adrian; NEUMAIR, Bernhard; FILSFILS, Clarence; Hegering, Heinz-Gerd; BRYSKIN, Igor; MCCABE, James D.; EVANS, John; STRASSNER, John; VIJAYANANDA, Kateel; SILVARAJAN, Kumar N.; MORROW, Monique; RAMASWAMI, Rajiv; ABECK, Sebastian; NADEAU, Thomas P. **Network Management:** Know It All. United States: Elsevier, 2009.

HASAN, Masum Z. **The Management of Data, Events, and Information Presentation for Network Management**. Disponível em <http://www.cs.uwaterloo.ca/research/tr/1996/39/rep.ps.Z>. Acesso em: 25 set. 2009.

HEWLETT PACKARD, **HP OpenView Integration Series: Application Style Guide**. Manufacturing Part Number: J1261-90006, Hewlett-Packard Company. United States, 2000.

HEWLETT PACKARD, **HP OpenView Integration Series: SNMP Developer's Guide**. Manufacturing Part Number: J1261-90008, Hewlett-Packard Company. United States, 2000.

INFRAX. **SNMP Protocol Reference**. Disponível em <http://www.infrax.com/en/network_protocols/snmp_protocol_reference.pdf>. Acesso em 14 nov. 2009.

INTERNATIONAL TELECOMMUNICATION UNION. **ITU-T M.3000**: Overview of TMN Recommendations. Genebra, Suíça, 1994.

INTERNATIONAL TELECOMMUNICATION UNION. **ITU-T M.3010**: Principles for a Telecommunications management network. Melbourne, Austrália, 1996.

INTERNATIONAL TELECOMMUNICATION UNION. **ITU-T M.3400**: TMN management functions. [S.l.] 1997.

KAY, Russell. **Event correlation.** Computerworld; Jul 28, 2003, pg 28.

KRISHNAMOORTHY C.S.; RAJEEV S. **Artificial Intelligence and Expert Systems for Engineers**. CRC Press. United States. 1996.

LANGER, Arthur M. **Analysis and Design of Information Systems**. Third Edition. Springer. 2007.

LARMAN, Craig. **Applying UML and Patterns**: An Introduction to Object-Oriented Analysis and Design and the Unified Process. Prentice Hall PTR: United States. 2002.

LENZ, Mario. AURIOL, Eric. MANAGO, Michel. **Diagnosis and Decision Support.** In:Lenz, M. Bartsch-Spörl, B. Burkhard H-D. Wess, S. (Eds.): Case-Based Reasoning Technology from foundations to Applications, Springer-Verlag, Heidelberg, Germany, 1998, p.51-90.

LUGER, George F. **Inteligência Artificial**: estruturas e estratégias para a solução de problemas complexos. São Paulo: Bookman. 2002.

MILLER, Mark A. **Managing Internetworks With Snmp**: The Definitive Guide to the Simple Network Management Protocol (SNMP) and SNMP version 2. 2ed, M & T Books, 1997.

NETWORK WORKING GROUP. **Request for Comments: 1155**: Structure and Identification of Management Information for TCP/IP-based Internets. [S.l.] 1990.

NETWORK WORKING GROUP. **Request for Comments: 1157**: A Simple Network Management Protocol (SNMP). [S.l.] 1990.

NETWORK WORKING GROUP. **Request for Comments: 3411**: An Architecture for Describing Simple Network Management Protocol (SNMP) Management Frameworks. [S.l.] 2002.

OBJECT MANAGEMENT GROUP. **Introduction to OMG's Unified Modeling Language.** 2005. Disponível em <http://www.omg.org/gettingstarted/what_is_uml.htm>. Acesso em 14/10/2009.

PROBST, Gilbert; RAUB, Steffen; ROMHARDT Kai. **Gestão do Conhecimento**: Os elementos construtivos do sucesso. Rio de Janeiro: Campus, 1997.

PRODANOV, C. **Manual de metodologia científica**. 3 ed. Novo Hamburgo, RS: FEEVALE, 2006.

RIESBECK, Christopher K; SCHANCK, Roger C. **Inside Case-Based Reasoning**. New Jersey: Lawrence Erlbaum, 1989.

RICHTER, Michael M. **Introduction**. In:Lenz, M. Bartsch-Spörl, B. Burkhard H-D. Wess, S. (Eds.): Case-Based Reasoning Technology from foundations to Applications, Springer-Verlag, Heidelberg, Germany, 1998, p.1-15.

RUSSELL, Stuart J.; NORVIG, Peter. **Inteligência artificial**: tradução da segunda edição. Tradução de PubliCare Consultoria. Rio de Janeiro. ed. Elsevier, 2004.

SHANNON, Claude E.; WEAVER, Warren. **A Mathematical Theory of Communication**. Illinois: University of Illinois Press, 1949.

SILVA, Chrystian V.; da. **Uma proposta de sistema de RBC para auxílio em atendimentos quiropráticos**. FEEVALE, Novo Hamburgo, 2008.

TANENBAUM, Andrew S.; **Redes de Computadores**: tradução da terceira edição. Ed Campus, Rio de Janeiro, 1997.

WANGENHEIM, Christiane Gresse von. WANGENHEIM, Aldo von. **Raciocínio baseado em casos**. Barueri, SP: Manole. 2003. 293p.

WATSON, Ian D. **Applying case-based reasoning**: techniques for enterprise systems. San Francisco: Morgan Faufmann Publishers, Inc, 1997.

WATSON, Ian. **CBR is a methodology not a technology**. University of Salford – UK, 2000. Disponível em <http://www.cs.auckland.ac.nz/~ian/papers/cbr/methodology.pdf>. Acesso em: 15 jun 2009.

WATSON, Ian. **A Distributed Case-Based Reasoning Application for Engineering Sales Support**. University of Auckland, New Zealand. Disponível em: <http://www.cs.auckland.ac.nz/~ian/papers/cbr/watson_ijcai99.pdf>. Acesso em: 15 jun 2009.

WIERINGA, R. J. **Design methods for reactive systems**: Yourdon, Statemate, and the UML. United States: Elsevier. 2003

YEATES, Donald; WAKEFIELD, Tony. **Systems Analysis and Design**. Second Edition. Finacial Times Prentice Hall. England. 2004.

YEN, David C.; DAVIS, William S. **The information system consultant's handbook**: systems analysis and design. CRC Press, United States. 1998.