UMA SOLUÇÃO PARA GERENCIAMENTO DE SISTEMAS EM LINUX.

Andrey Soares
Universidade Federal de Santa Catarina – UFSC
Campus Universitário – Trindade – Florianópolis, SC – 88040-900 - andrey@inf.ufsc.br

Marcello thiry, Dr.
Universidade do Vale do Itajaí – UNIVALI Campus São José
Campus VII - Rodovia SC 407 Km 4 – São José, SC – 88122-00 - thiry@sj.univali.br

Abstract

This work consists of a study of a model of network management using the SNMP architecture. The SNMP architecture is divided into Agent and M anagement applications, and in this paper the application called Agent will receive more attention..

The aspects worked in this paper start with the management functions and the necessity of MIB until the MIB definition and criation.

The result, was the MI B definition that has the objective of storaging information about the hardware and Software inventary at workstations.

It is also is presented the next steps of this work, where will be used the XML as language contents, where the objective is mapping MIB to XML.

Keywords: Linux, Systems management, MIB, XML

1 INTRODUÇÃO

Com o contínuo aumento da tecnologia, cada vez mais empresas estão investindo em redes de computadores que se tornam maiores e com uma grande quantidade de equipamentos conectados. Atualmente, uma infra-estrutura de rede é composta por várias redes locais (LANs) ou até mesmo, por redes de longa distância (WANs), incluindo equipamentos como roteadores, pontes, PCs, servidores, terminais e outros periféricos.

Segundo PEITER (2000), embora j á existam MIBs (*Management Information Base*) prontas com agentes e gerentes implementados que monitoram uma grande parte de informações de interesse de uma arquitetura de rede, é difícil achar uma que se adapte as nossas necessidades. Existem várias implem entações SNMP sendo difícil conseguir uma aplicação que resolva os problemas específicos de cada empresa. Em uma universidade por exemplo, vários alunos diferentes utilizam os mesmos computadores e com a facilidade de instalar softwares piratas ou pela *Internet* ou por outros meios, pode acarretar em multas para a instituição devido ao número de *softwares* sem licença de uso. Estes *softwares* podem também causar a diminuição do espaço de disco, ou falta de memória, sobrecarregando o computador de alguma maneir a que acarretará em outros tipos de problemas.

Sugerimos neste trabalho, uma MIB específica para inventariar *hardwares* e *softwares* de um computador em plataforma Linux, onde um agente fará a coleta de informações no sistema e disponibilizará estas informações através de uma Base de Informações - MIB.

2 GERÊNCIA DE SISTEMAS

Dentre as atividades de gerência e mais especificamente dos objetos gerenciados em uma rede de computadores, pode-se citar as estações de trabalho, que demandam uma atenção especial por estarem sendo operadas pelos usuários. Ao contrário de alguns objetos gerenciados, como um *Hub*, que tem uma função específica e é geralmente gerenciado sob um mesmo enfoque, temos a estação de trabalho que permite diversas utilizações, pois dependem das necessidades dos usuários e consequentemente apresentam diversos enfoques de gerência. Desta forma, pode-se analisar a utilização de uma estação, verificar quais atividades podem influenciar o andamento das operações da rede, como por exemplo, se um usuário estiver fazendo um *download* de um arquivo grande durante o horário comercial de uma empresa, poderá estar influenciando na performance da rede com o aumento do tráfego.

As cinco áreas funcionais ISO (*International Organization for Standardization*), separadas devido a necessidade no gerenciamento OSI (*Open System Interconection*), e que geralmente são usadas em redes de computadores, podem ser adaptadas ao gerenciamento de sistemas (HEGERING, 1994) (HELD, 1996).

- **Contabilização**: pode indicar quando um usuário efetua *logon* na rede ou quais serviços foram usados, tais como ftp, telnet, http, etc;
- **Configuração**: identificar qual *hardware* e *software* está instalado na máquina;
- **Segurança**: quando ocorreu uma tentativa de acesso indevido, quando um comando foi executado sem os privilégios do *root*, identificar padrões de ataque;
- **Desempenho**: quanto de cpu, memória ou espaço em disco estão sendo ocupados;
- **Falha**: quando e onde um problema ocorreu, notificar o gerente, da detecção de uma falha e fornecer informações suficientes para que sejam tomadas as providências cabíveis, procurar e prevenir pontos vulneráveis de falhas, registrar os problemas (e suas soluções) ocorridos para consultas futuras.

2.1 Funções de Gerenciamento

Estas funções são citadas como de grande importância no apoio às atividades do administrador de sistemas (SOARES et. al., 2001):

- **Inventário de Hardware e Software** : prover funcionalidades para adquirir informações sobre o hardware e software instalado nos computadores clientes. O agente deverá coletar informações de: CPU, Memória, Pontos de Montagem, Interrupções, Porta I/O, PCI, Canais DMA, Placa de Vídeo, Monitor, Teclado, Mouse, Impressora, Unidades de Disco, Protocolos, Placa de Rede, Cd-Rom, *Floppy*, Console, Portas COM / LPT, Modem, Placa de Som, etc.

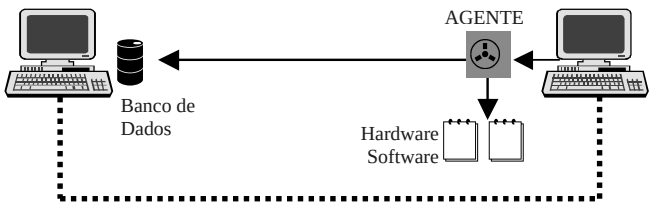

Figura 1: Esquema do agente de Inventário de Hardware e Software

- **Instalação e distribuição de software** : prover funcionalidades para:
  - Instalar e remover software,
  - Atualizar o Sistema Operacional,
  - Executar tarefas administrativas como por exemplo verificação de disco,
  - Criar status da execução,
  - Registro das instalações (data, hora, etc.),
  - Os clientes não podem modificar as configurações dos agentes,
  - Verificação de espaço em disco.

- **Medição de software** : prover funcionalidades para med ir a utilização de programas e gerenciar licenças. Esta funcionalidade possibilita o controle mais intenso sobre os programas utilizados nos computadores clientes, tais como:
  - Número de licenças usadas,
  - Registro da execução de programas (rede, disco ou di squete),
  - Autorização de uso de programas,
  - Percentual de utilização de programas,
  - Tempo de utilização dos programas,
  - Emissão de alerta para a utilização de *software* sem licença,
  - Controle de atualização de *software*.

- **Registro de problemas** : prover funciona lidades para diagnosticar problemas de *hardware*, *software* ou rede. Na gerência de sistemas, muitos problemas se repetem, surgindo então a necessidade de se conhecer como um determinado problema foi resolvido. Para isto deve ser criado um sistema que possib ilite armazenar os problemas e suas respectivas soluções, constituindo -se assim uma memória histórica composta das experiências e conhecimentos utilizados na solução dos problemas.
  A resolução de problemas consiste na verificação dos arquivos de Log para verificar quais erros, alertas ou condições críticas ocorreram e procurar na base de casos os casos mais similares.

- **Monitoração do sistema**: podem ser executados a partir do computador dos técnicos para obter informações em tempo real sobre os computadores clientes; Constitui-se de um agente que coleta, de tempos em tempos, informações referentes ao sistema (Figura 2) e envia as informações coletadas para o banco de dados do sistema. Deste modo será possível manter um histórico do desempenho do sistema e o mesmo poderá ser verificado através de gráficos. As informações coletadas são:
  - **CPU**: Ocupada com o Usuário, Ocupada com o Sistema
  - **Memória**: Usada e Livre
  - **Swap**: Usado e Livre
  - **Disco**: Usado e Livre
  - **Processos**: Dormindo, Rodando, Parado, Zumbi
  - **Rede**: Pacotes Recebidos e Transmitidos

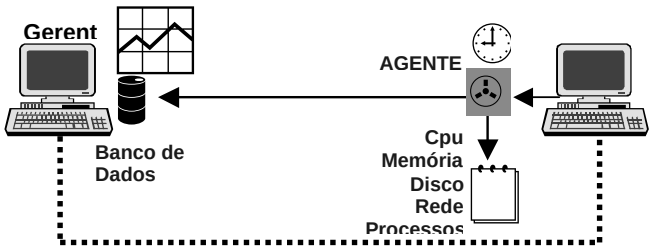

Figura 2: Esquema de monitoramento do sistema

- **Descoberta e reconhecimento da Rede** : É a utilização de métodos de descoberta de elementos de rede com localização e coleta de informações, como por exemplo:
  - Método baseado em endereço IP, onde o agente de descoberta envia uma mensagem em *broadcast* e a partir da resposta, vai mapeando os elementos;
  - Método baseado em *logon*, onde após o usuário efetuar e validar o *logon*, o servidor envia uma mensagem para o cliente informando da sua existência na rede;
  - Método de reconhecimento**,** onde o agente de inventário de *hardware* e *software* após executar a coleta de dados terá as seguintes informações: endereço IP, nome da máquina, rota padrão, sistema operacional, versão do sistema, etc.

- **Ferramentas remotas**: permitem que os técnicos controlem o computador cliente como se estivessem no local, através de chat com o usuário, cópia de arquivos para o computador cliente, reiniciar o computador ou iniciar programas e serviços. Estas ferramentas tem colaborado na eficiência de diagnósticos e na tomada de decisão, antes mesmo que o usuário tomasse conhecimento do problema ocorrido;

- **Monitor de rede** : permite um administrador capturar e seguir o tráfego para o computador cliente ou a partir do computador cliente. Em adicional o monitor de rede inclui monitoramento de eventos e análise dos dados capturados.

3 BASE DE INFORMAÇÕES - MIB

Dentre as funções de gerenciamento citadas acima, verificou-se que as funções **Inventário de *hardware* e *software*** e **Monitoração do sistema** , poderiam ser melhor disponibilizadas através de uma MIB.

Uma MIB (MILLER, 1999) é uma estrutura que contém as variáveis necessárias para monitorar, gerenciar ou administrar os componentes em redes Internet. Basicamente, existem três tipos de MIBs (PEITER, 2000): MIB II, MIB experimental e MIB privada. A MIB I se tornou obsoleta quando foi acrescentada de alguns itens, tornando-se assim a MIB II. A MIB II fornece informações sobre o equipamento gerenciado como por exemplo, o estado da interface, informações sobre os protocolos de rede, número de pacotes com erros, etc. As MIBs experimentais são aquelas que estão em fase de testes para que no futuro possam ser padronizadas. As MIBs privadas são específicas dos equipamentos gerenciados, como *HUBs, Swiches,* Roteadores, etc. Estas fornecem informações particulares de cada um destes equipamentos.

As informações de uma MIB são armazenadas de uma forma padronizada e organizada para que seja fácil acessá-las quando necessário. As informações da MIB-II são agrupadas por

protocolos (incluindo TCP, IP, UDP, SNMP, e outros) e outras categorias, incluindo "sistemas" e "interfaces".

Os objetos gerenciados são especificados usando uma linguagem própria e não ASN.1 (*Abstract Syntax Notation One*) puro (MILLER, 1999). A linguagem utiliza os módulos ASN.1 que definem macros, descrições textuais modificando e qualificando as macros e elementos de ASN.1, e outros itens de uso comum.

O gerenciamento SNMP (*Simple Network Manage ment Protocol*) é utilizado para gerenciar muitas outras coisas além de roteadores, hubs, computadores e outros dispositivos de interconexão de redes, como por exemplo uma MIB para gerenciar dispositivos do tipo UPS (*Uninterruptible Power Supply*), descrita na RFC 1628. Assim, é possível encontrar dispositivos contendo agentes SNMP que não implementam alguns grupos da MIB -II, bem como encontrar dispositivos que implementem outras informações de gerenciamento (WEBBER, 1997).

Desta forma, para utilizarmos as in formações de gerenciamento sugeridas nas funções de gerenciamento escolhidas, deve-se criar uma MIB que suporte as informações desejadas ou então utilizar uma MIB já disponível. Neste caso, utilizamos a *Network Client Application Mib* proposta por WEBBER (1997) por possuir objetos identificados, que melhor atendem estas necessidades.

3.1 A arquitetura de gerenciamento - SNMP

A arquitetura de gerenciamento SNMP consiste em várias entidades que trabalham individualmente executando suas funções, e que através de um protocolo de comunicação podem enviar suas informações a outras entidades que permitam mostrá-las a um usuário.

As implementações básicas do SNMP permitem monitorar e isolar falhas, já as aplicações mais sofisticadas possibilitam gerenciar o desempenho e a configuração da rede. Estas aplicações, em geral, incorporam *menus* e alarmes para facilitar a interação com o usuário gerente de rede.

A arquitetura SNMP consiste em (PERKINS, 1997):

- um ou mais nodos onde cada qual possui uma entidade de processamento chamada agente;
- pelo menos uma estação gerente contendo um ou mais entidades de processamento chamada aplicação de gerenciamento ou somente gerente;
- opcionalmente as entidades de processamento podem fazer as duas coisas, ser agente e gerente ao mesmo tempo, e são chamadas de *dual-role entity*;
- informações de gerenciamento em cada nodo gerenciado que descrevem o estado da configuração, estatísticas e que controlam as ações do nodo gerenciado;
- um protocolo de gerenciamento, o qual os gerentes e agentes utiliz am para trocar mensagens de gerenciamento.

A Figura 3 ilustra um modelo de gerenciamento de rede SNMP.

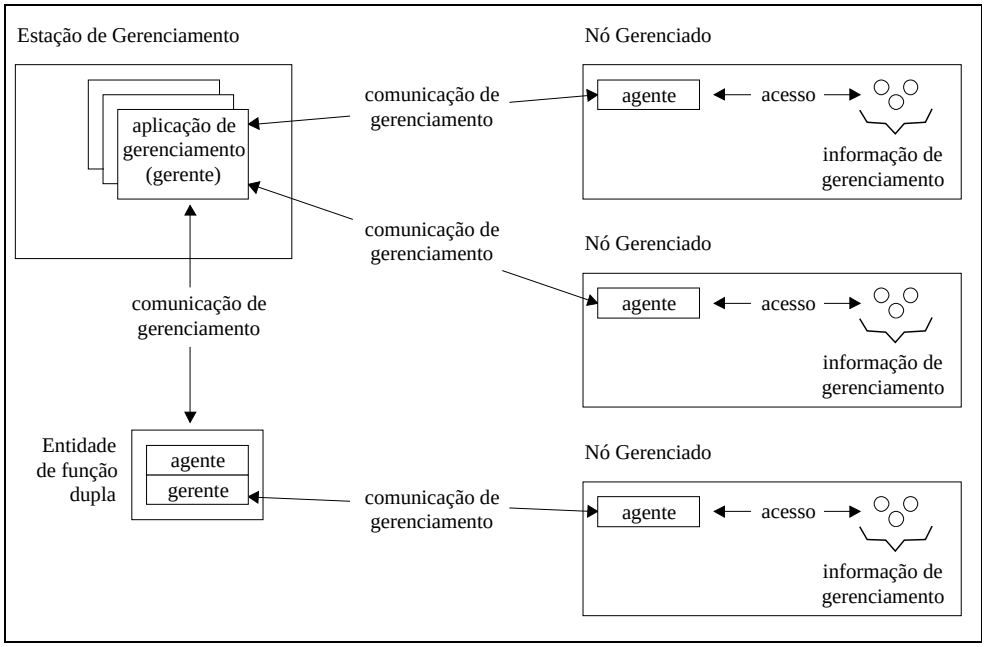

Figura 3: O modelo SNMP em uma rede gerenciada. (PERKINS, 1997)

As estações agente são as estações que coletam informações, trabalhando localmente nos nodos gerenciados e armazenam essas informações em uma base de dados local chamada MIB. A estação gerente mostra ao usuário a situação de cada nodo da rede, através de pedidos enviados pelo protocolo SNM P ao agente, que coleta as informações pedidas na sua MIB e responde para o gerente. A estação agente também pode enviar informações de gerenciamento para um gerente sem ser solicitado. Normalmente, isso ocorre devido a alguma irregularidade ocorrida na estação.

Um gerente SNMP possui também uma interface com o usuário e é uma entidade capaz de gerar e receber pedidos e modificar as informações de gerenciamento. A estação de gerenciamento deve possuir, no mínimo (STALLINGS, 1993):

- um conjunto de aplicações para análise de dados, recuperação de falhas, etc.;
- uma interface pela qual o gerente da rede possa monitorar e controlar a rede de computadores;
- a capacidade de traduzir os requisitos de gerente de redes em monitoração e controle efetivos dos elementos remotos na rede;
- uma base de dados de informações extraídas das MIBs de todas as entidades gerenciadas na rede.

A estação agente é o elemento ativo do sistema de gerenciamento de redes. Os nodos da rede como pontes, *hubs*, roteadores, dentre outros, têm que p ossuir o agente SNMP para que possam ser monitorados pelas estações gerente. O agente tem a função de receber, processar e responder os pedidos das estações gerente e, às vezes, enviar informações não solicitadas. Um agente deve ter acesso à base de inform ações gerenciadas para poder responder os pedidos feitos pelo gerente, e deve ser notificada de eventos internos para poder gerar informações e enviá -las a um gerente determinado.

Em uma MIB, os objetos gerenciados possuem atributos que contém determinadas informações. A descrição dos objetos é dividida em cinco partes: o nome do objeto, a sintaxe abstrata do objeto, a descrição textual do significado do objeto, o tipo de acesso permitido ao objeto, e o estado do objeto. A Figura 4 apresenta a descrição do objeto *redeUsuarioAtual*.

```
redeUsuarioAtual OBJECT-TYPE
      SYNTAX  DisplayString (SIZE (0..255))
      ACCESS  read-only
      STATUS  mandatory
      DESCRIPTION
              "Nome de login do usuario que está
               atualmente utilizando o computador "
      ::= { rede 1 }
```

Figura 4: Objeto *redeUsuarioAtual* da MIB-II

O nome do objeto é um nome textual para o tipo de objeto, denominado "Descritor de Objetos" que acompanha o identificador de objeto . A sintaxe abstrata diz respeito ao tipo de valor que o objeto armazena. A descrição textual é vital para que os objetos tenham significados consistentes. É uma descrição da semântica do objeto. O tipo de acesso permitido ao objeto pode ser para leitura, leitura e escrita ou inacessível. E por fim o estado do objeto pode ser obrigatório, opcional ou obsoleto.

Os requisitos para desenvolver um agente extensível MIB são (PEITER, 2000):

- ter instalado o protocolo TCP/IP;
- instalar o serviço SNMP, pois é o serviço SNMP que irá monitorar a porta 161 UDP para verificar a chegada de mensagens SNMP;
- configurar o serviço SNMP para que as mensagens recebidas e enviadas estejam de acordo com as definições do administrador de rede;
- desenvolver uma biblioteca de funções que processe mensagens *Get Request*, *Get-Next Request*, *Set Request* sobre os dados definidos em uma MIB;
- instalar a biblioteca de funções do agente extensível para que na inicialização do serviço SNMP a biblioteca seja carregada para a memória junto com o serviço SNMP.

4 MIB PROPOSTA

Quando abordamos um novo projeto de MIB, é importante ter-se em mente algumas categorias de objetos, para pensar no problema de uma forma organizada (PERKINS, 1997 *apud* WEBBER, 1997):

- **Ações**: controlam um sistema. Objetivos do tipo ação são usados para permitir a execução de tarefas bem definidas, como: reinicializar um dispositivo, desativar um serviço de rede, etc.;
- **Estatísticas**: fornecem informação útil sobre o que aconteceu no sistema desde o início de um certo intervalo de tempo. Estatísticas podem incluir itens como: o número de pacotes transmitidos por uma interface de rede, ou o número de vezes que um usuário se conectou a uma máquina.
- **Estados**: indicam o estado corrente de um sistema. Estado podem incluir itens como: uma placa está inicializando, ou um ar-condicionado está com defeito;
- **Componentes**: ajudam a descrever conjuntos de dispositivos físicos e lógicos, ou serviços que estão sob controle do agente SNMP. Por exemplo, as placas presentes em um sistema de chassis com múltiplos *slots*, ou ainda os nomes dos serviços ativos em um servidor de arquivos;
- **Atributos**: são as propriedades de um objeto modelado, que descrevem coisas relativamente estática sobre um dispositivo ou serviço. Exemplo: o número de portas em

um *hub Ethernet*, ou a pessoa a chamar em caso de falha de um dispositivo, ou ainda que tipo de CPU está instalada no sistema.

Estando os objetos agrupados em suas categorias correspondentes, estes podem ser agrupados em uma tabela -resumo, que contém uma linha para cada componente indicado, onde os sub -componentes devem estar nas linhas que o seguem. As colunas pertencentes a uma linha são: Componente, Cardinalidade, Atributos, Estatísticas e Estados.

Os passos necessários para projetar uma MIB, são apresentados na Figura 5, onde em cada estágio, um conjunto de documentos precisa ser produzido.

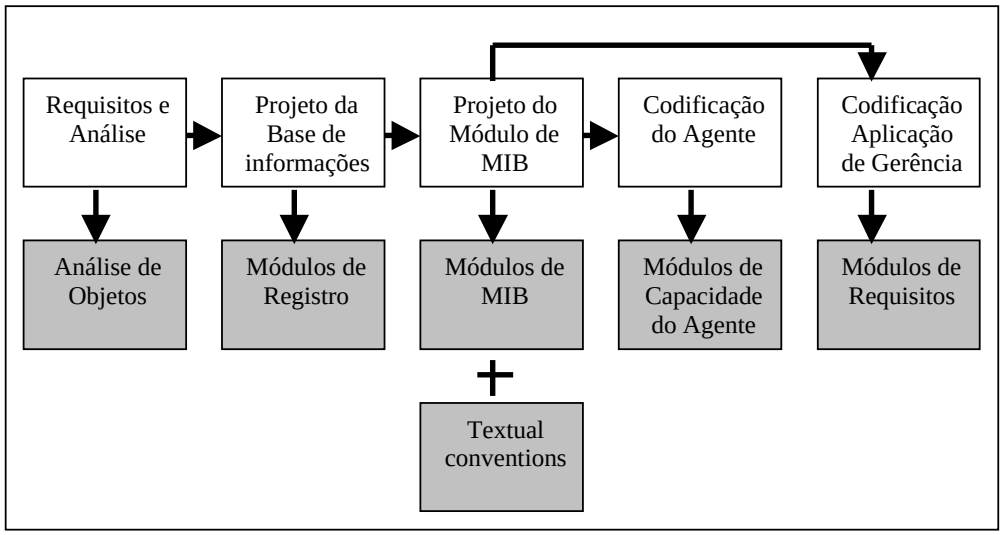

Figura 5: Passos no Desenvolvimento de uma MIB (WEBBER, 1997)

A primeira fase, descreve os objetos gerenciáveis que serão modelados na MIB; a segunda fase consiste em estabelecer a base geral de informações, e decidir como organizar as definições de objetos em um mais módulos de MIB; a terceira fase é onde são criadas as definições de objetos gerenciados. As duas últimas fases produzem módulos de informação que consistem em respostas à MIB recém-criada.

Segundo WEBBER (1997), antes de definir a MIB, é necessário decidir qual a versão de SMI (*Structured of Management Information*) SNMP deve ser utilizada, sendo adotado para este módulo de MIB a sintaxe da SMIv2.

A MIB definida (Figura 6) tem o objetivo de monitorar as estações de trabalho, fazendo a coleta de informações de *Hardware* e *Software*, constituindo-se assim em um registro de inventário.

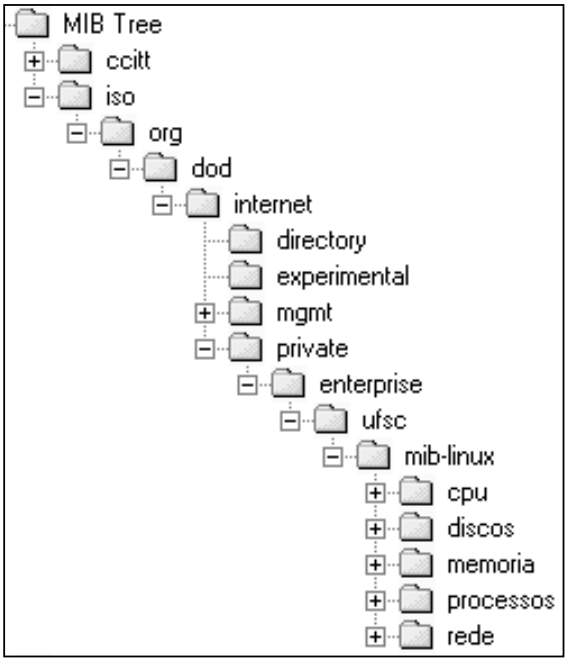

Figura 6: Grupo de objetos da MIBLINUX

Os objetos identificados para esta MIB, estão agrupados em 5 categorias:

- **CPU**: podemos verificar o fabricante, o tipo e a velocidade do processador;
- **DISCOS**: informações referente aos *drives* instalados. Podemos saber a letra da unidade, espaço total, espaço livre, o nome do volume, o número de série, o sistema de arquivos, etc de cada uma das unidades de disco individualmente;
- **MEMÓRIA**: dados sobre a memória do computador. Por exemplo, pod emos saber a porcentagem utilizada, o total e a quantidade livre de memória;
- **PROCESSOS**: são as informações dos processos que estão sendo executados no computador no exato momento. Podemos saber o numero total de processos, o número de identificação do proc esso, o nome do processo, o tipo ou classe do processo e podemos alterar o status do processo de *Running* para *Terminated*. Se a variável processStatus for alterada para o valor igual a 1 o processo referente será finalizado;
- **REDE**: podemos verificar o nome do usuário logado no sistema, o nome do computador e o nome do domínio em que o usuário está logado e o endereço IP.

A seguir será apresentado o grupo Memória em formato ASN.1:

```
 memUtilizacao OBJECT-TYPE
        SYNTAX INTEGER
        ACCESS read-only
        STATUS mandatory
        DESCRIPTION
                  "Porcentagem de utilizacao da memoria RAM."
        ::= { memoria 1}

   memFisicaTotal OBJECT-TYPE
        SYNTAX INTEGER
```

```
            ACCESS read-only
            STATUS mandatory
            DESCRIPTION
                        "Total de memoria Fisica ( RAM )."
            ::= { memoria 2}

      memFisicaLivre OBJECT-TYPE
            SYNTAX INTEGER
            ACCESS read-only
            STATUS mandatory
            DESCRIPTION
                        "Quantidade de memoria fisica livre ( RAM )."
            ::= { memoria 3}
```

5 CONCLUSÕES E PRÓXIMO S PASSOS

A utilização da MIB SNMP para o gerenciamento de sistemas, tem sido um padrão para controlar e gerenciar vários elementos de rede s. Sendo as comunicações entre gerentes e agentes, estabelecidas através das primitivas SNMP: GET e SET, o que se constitui de um fator limitante na manipulação de informações.

Durante este projeto, verificou -se a possibilidade de usar XML como ferramenta para manipulação de informações de uma MIB, sendo destacado a criação e manipulação de conteúdos dinâmicos.

A próxima etapa deste trabalho é a utilização de XML como agentes de busca de informações, onde o agente XML faria o acesso as informações da MIB de modo transparente para o gerente (Figura 7).

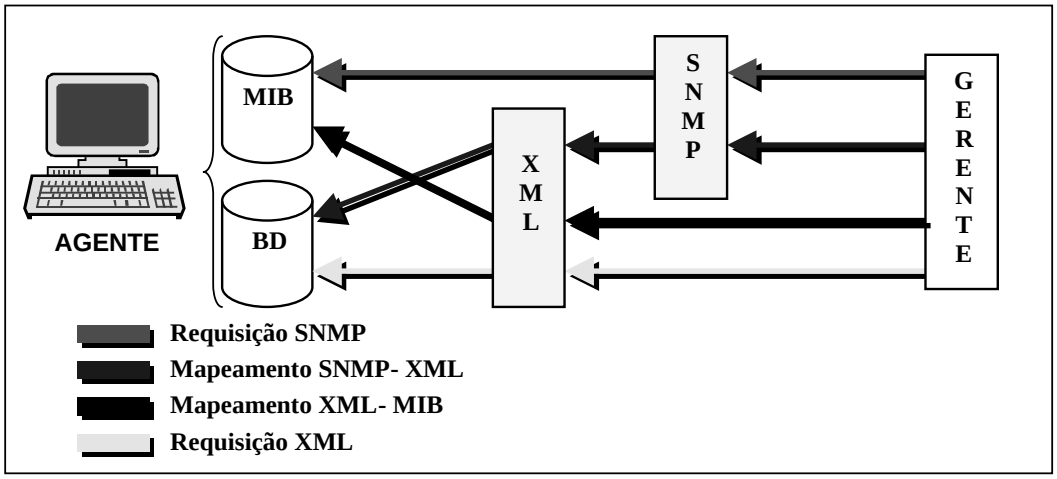

Figura 7: Esquema de mapeamento MIB- XML

Serão apresentadas 4 possibilidades de gerenciar uma Base de Informações Gerenciais (MIB), sendo esta padronizada ou não.

1. Um software de gerenciamento, requisita informações da MIB através de primitivas SNMP. Método tradicional de gerenciamento.
2. O software de gerenciamento incorpora as informações, mesmo sem possuir um MIB padrão. Onde o gerente faz requisições SNMP e estas são repassadas ao agente XML que se encarrega de obter as informações.

3. Existe uma MIB padrão coletando informações, porém não existe um gerente que faça requisições SNMP. Neste caso, o agente XML converte as solicitações em SNMP.
4. Existe um conjunto de informações de gerência, porém não existem uma MIB padrão coletando estas informações e nem um software de gerenciamento padrão.

Desta forma a base de dados poderia ser implementada em qualquer ferramenta, pois o agente XML manteria um mapeamento (Repr esentação dos Objetos Gerenciáveis) das informações MIB que seriam conhecidas dos gerentes. Isto permitiria que fosse mantida uma base de informações completa, onde caberia ao agente apresentar as informações no formato de MIB, ou seja, a partir de uma mesma base de dados, pode-se simular várias MIBs.

6 BIBLIOGRAFIA

HEGERING, Heinz-Gerd, ABECK, Sebastian. **Integrated Network and System Management** . Addison-Wesley Publishing Company. Munich, 1994.

HELD, Gilbert. **Lan Management with SNMP and RMON** . Wiley Compute r Publishing. New York. 1996.

HOLZNER, Steven. Inside XML. New Riders, 2001.

MILLER, Mark. **Managing Internetworks with SNMP** . IETF Network Management Documents. M&T Books, 1999.

PEITER, Rui C. **Um modelo de um agente SNMP para gerenciamento de rede** . TCC- Univali-Ciência da Computação, 2000.

PERKINS, David; McGinnis, Evan. **Undestanding SNMP MIBs**. Prentice-Hall PTR, 1997.

SOARES, Andrey, THIRY, Marcello. **Uma contribuição para o gerenciamento de sistemas: Estudo de caso em Linux**, CPGCC-UFSC, 2001.

STALLINGS, W., SNMP, SNMPv2, and CMIP **. The practical Guide to Network -Management Standards**. Addison Wesley, 1993.

WEBBER, Celso K. **Uma Mib para aplicações internet** . Dissertação de Mestrado. CPGCC - UFSC, 1997