

ANTONIO RAFAEL SANT'ANA

ÁRVORE HIPERBÓLICA COMO FERRAMENTA DE MONITORAMENTO DE REDE

Monografia de graduação apresentada ao Departamento de Ciência da Computação da Universidade Federal de Lavras como parte das exigências do Curso de Ciência da Computação para obtenção do título de Bacharel em Ciência da Computação.

LAVRAS
MINAS GERAIS – BRASIL
2008

ANTONIO RAFAEL SANT'ANA

ÁRVORE HIPERBÓLICA COMO FERRAMENTA DE MONITORAMENTO DE REDE

Monografia de graduação apresentada ao Departamento de Ciência da Computação da Universidade Federal de Lavras como parte das exigências do Curso de Ciência da Computação para obtenção do título de Bacharel em Ciência da Computação.

Área de concentração:

Redes de Computadores, Interface Homem Máquina.

Orientador:

Prof. Luiz Henrique Andrade Correia

LAVRAS
MINAS GERAIS – BRASIL
2008

**Ficha Catalográfica preparada pela Divisão de Processo Técnico da Biblioteca
Central da UFLA**

Sant'Ana, Antonio Rafael

Árvore Hiperbólica com Ferramenta de Monitoramento de Rede / Antonio Rafael
Sant'Ana – Minas Gerais, 2008.

Monografia de Graduação – Universidade Federal de Lavras. Departamento de Ciência da
Computação.

1. Monitoramento de Redes. 2. Árvore Hiperbólica. I. SANTANA, A. R. II. Universidade
Federal de Lavras. III. Título.

ANTONIO RAFAEL SANT'ANA

ÁRVORE HIPERBÓLICA COMO FERRAMENTA DE MONITORAMENTO DE REDE

Monografia de graduação apresentada ao Departamento de Ciência da Computação da Universidade Federal de Lavras como parte das exigências do Curso de Ciência da Computação para obtenção do título de Bacharel em Ciência da Computação.

Aprovada em ___ de _____ de 2008

Prof. Bráulio Adriano de Mello

Prof. Reginaldo Ferreira de Souza

Prof. Luiz Henrique Andrade Correia
(Orientador)

LAVRAS
MINAS GERAIS – BRASIL
2008

Dedico este trabalho, primeiramente, aos meus pais pelo esforço que fizeram para que eu atingisse meus objetivos. A eles agradeço pelo que sou hoje. Ao meu Mestre Professor Luiz Henrique Andrade Correia, aos meus colegas de trabalho, em especial Erasmo Evangelista de Oliveira e Anderson Bernado dos Santos que me ensinaram muito durante toda minha vida acadêmica. E, finalmente, dedico a todos que me fortaleceram e me tornaram competente para realizar todo meu trabalho acadêmico.

Agradecimentos

Agradeço aos meus pais pela oportunidade que me deram para retribuir tudo o que me ensinaram, aos meus amigos pelo desafio de sempre aprender a acreditar e ter confiança, pois amigos apenas são e a todos os colegas e familiares que me apoiaram nesta jornada.

Agradeço, em especial:

A Erasmu Evangelista de Oliveira e Anderson Bernado dos Santos, com quem aprendi muito durante os anos de luta na universidade;

A Jaime Daniel Correia Mendes, Claudiane Maria Oliveira e Thiago do Prado Ramos, que sempre foram parceiros na jornada acadêmica e com certeza ficará uma eterna amizade;

Aos Professores da Universidade Federal de Lavras, em especial, Professor Luiz Henrique Andrade Correia e aos Funcionários da Universidade Federal de Lavras, em especial, Angela Maria Silva Gonçalves e Deivson Henrique Rodrigues de Almeida;

Aos meus amigos, familiares, colegas e conhecidos que aqui não mencionei e que sabem que tiveram sua importância na minha vida.

Sinceros Agradecimentos.

ÁRVORE HIPERBÓLICA COMO FERRAMENTA DE MONITORAMENTO DE REDE

RESUMO

A partir dos avanços tecnológicos proporcionados principalmente pelas tecnologias da informação e de comunicação, tornou-se indispensável a diversas organizações a utilização de redes de computadores. Em grande parte destas instituições existe um grande número de equipamentos de redes que são de vital importância para o desempenho de suas atividades. Na sua maioria essas atividades são críticas e necessitam de alta disponibilidade de seus equipamentos, que devem ser constantemente monitorados e gerenciados. Atualmente instituições e empresas têm utilizado várias ferramentas no gerenciamento de seus sistemas de informação. Uma alternativa tem sido a utilização de um programa de código livre, denominado Nagios. Este programa disponibiliza um módulo gráfico para monitoramento de toda a estrutura hierárquica da rede de computadores. A proposta deste trabalho é a geração de uma estrutura gráfica mais adequada ao programa, sendo que a atual não apresenta dinamicidade e interatividade adequadas às grandes estruturas hierárquicas. Para cumprir tal objetivo, esse trabalho apresenta uma estrutura de árvore hiperbólica, uma técnica de visualização de grandes hierarquias, como um módulo de monitoramento para o Nagios. Esse módulo tem como objetivo a melhora da usabilidade e conseqüentemente do processo de administração dos serviços de rede monitorados pelo software Nagios.

Palavras-chave: monitoramento de rede, árvore hiperbólica, Nagios, visualização de grandes hierarquias, usabilidade.

HIPERBOLIC TREE AS NETWORK MONITORING TOOL

ABSTRACT

From the technological progresses mainly provided by the information technology, it has become indispensable the use of computer networks for the organizations. In the greater number of these organizations, there are several network equipments which are very important for the acting of their activities. The majority of those activities is critical and need to have high availability of the equipments that also should constantly be monitored and managed. Nowadays, institutions and companies have used several tools to administrate and monitor their information systems. An alternative has been the use of the free code software denominated Nagios. That software has a graphic module to monitor the whole hierarchical structure of a computers network. The proposal of this paper is to improve this graphical structure generated by the responsible module, taking into consideration that the current does not have dynamism and interaction adequate for large hierarchical structures. To accomplish that goal, this paper introduces a hyperbolic structure tree, a visualization technique of large hierarchies, as a monitoring module for the Nagios. That module tries to improve the usability of the software Nagios, optimizing the administration process of the monitored network services.

Key-Words: network management, hyperbolic tree, Nagios, visualization of large hierarchies, usability.

SUMÁRIO

LISTA DE FIGURAS

1. INTRODUÇÃO	1
2. GERENCIAMENTO DE REDES	4
2.1. Arquitetura de Gerenciamento de Redes	5
2.2. Protocolos de Gerenciamento	8
2.3. Áreas funcionais do gerenciamento de redes	9
2.3.1. Gerenciamento de falhas	9
2.3.2. Gerenciamento de contabilização	10
2.3.3. Gerenciamento de desempenho	10
2.3.4. Gerenciamento de configuração	11
2.3.5. Gerenciamento de segurança	11
2.4. Software de monitoramento integrado Nagios	12
3. VISUALIZAÇÃO DE INFORMAÇÃO E USABILIDADE DE SOFTWARE	15
3.1. Usabilidade de software	15
3.2. Introdução à visualização de informações	17
3.3. Visualização de estruturas hierárquicas	18
3.4. Visualização de informações de contexto e detalhe	20
3.4.1. Abordagem Visão Geral + Detalhe	20
3.4.2. Abordagem Foco + Contexto	23
3.5. Árvore Hiperbólica, HiperNavegador e HiperEditor	25
3.5.1. HiperNavegador	25
3.5.2. HiperEditor	26
4. METODOLOGIA	28
4.1. Materiais	28
4.2. Modelo de geração dinâmica da árvore hiperbólica	30
5. RESULTADOS E DISCUSSÃO	33
5.1. Modelo de visualização <i>balanced tree</i>	33
5.2. Modelo de visualização circular	34
5.3. Modelo de visualização: árvore hiperbólica	36
5.4. Melhorias na usabilidade do Nagios	37
6. CONCLUSÃO	39
6.1. Trabalhos futuros	39
REFERÊNCIAS BIBLIOGRÁFICAS	41

LISTA DE FIGURAS

Figura 1 - Camadas de uma arquitetura de gerenciamento.....	5
Figura 2 - Modelo de gerenciamento centralizado.	7
Figura 3 - Modelo de gerenciamento distribuído	7
Figura 4 - Geração gráfica da hierarquia de rede.....	13
Figura 5 - Interface WAP.	14
Figura 6 - Modelo de Referência para Visualização.	17
Figura 7 - Digrama <i>top-down</i>	18
Figura 8 - Listas delimitadas.	19
Figura 9 - Listas endentadas.....	20
Figura 10- Conjuntos aninhados.....	20
Figura 11- Microsoft Explorer Browser.....	22
Figura 12- Interface Visão Geral+Detalhe utilizada na exibição de mapas.....	22
Figura 13 - Navegador Hiperbólico.	24
Figura 14 - Esquema HiperEditor x HiperNavegador.....	27
Figura 15 - Navegador Hiperbólico x Nagios.....	32
Figura 16 - Modelo de visualização <i>balanced tree</i>	34
Figura 17 - Modelo de visualização circular <i>marked up</i>	35
Figura 18 - Visualização da árvore hiperbólica com nó raiz no centro.....	36
Figura 19 - Visualização com nó valsa.ufla.br no centro.....	37
Figura 20 - Exemplo de visualização da árvore hiperbólica.....	38

1. INTRODUÇÃO

Para atender a grande quantidade de informações geradas com o crescimento das corporações é necessário que a estrutura de Tecnologia da Informação (TI) da empresa esteja preparada para comportar esse crescimento. A disponibilidade das informações geradas pela empresa pode influenciar de maneira decisiva em sua competitividade frente aos seus concorrentes. Para garantir o fluxo das informações que trafegam tanto internamente, através de sua rede interna, quanto externamente, através da *web*, é necessário que os equipamentos de rede dessa empresa sejam monitorados constantemente para evitar que ocorram problemas no processo do ciclo da informação [1].

Além disso, a existência de vários tipos de dispositivos ligados à rede, que faz necessária a criação de um esquema que ofereça soluções de gerenciamento adaptáveis e estruturadas, permitindo a monitoração de equipamentos compatíveis ou não [2].

Os fabricantes de determinados equipamentos de rede disponibilizam soluções de gerenciamento proprietárias. No entanto esse gerenciamento não se estende a equipamentos de outros fabricantes. Dessa forma, em uma organização onde o número de equipamentos ligados em rede é consideravelmente grande, e os equipamentos de rede são diversificados, certamente haverá dificuldade no gerenciamento, sendo necessário a utilização de vários softwares. Para contornar essa situação é utilizado o gerenciamento integrado que tem como objetivo utilizar apenas uma ferramenta de gerenciamento para extrair informações de toda a rede de comunicação [1].

Existem vários programas no mercado utilizados para o gerenciamento e monitoramento integrado da rede de computadores. Porém, cada vez mais soluções com software livre são utilizadas devido à popularização e melhoria da documentação de várias ferramentas. Como exemplo dessas ferramentas de código aberto podemos citar: MRTG [3], RRDTool [4], Nagios [5], Cacti [6], Zabbix [7], Munin [8], Zenoss [9] e Nav [10]. O software que escolhido para este trabalho foi o Nagios cujo principal vantagem está na robustez e grande popularidade [11].

Esse software possui vários recursos de monitoramentos, mas deixa muito a desejar quando se trata de exibir a hierarquia da rede que está sendo monitorada. A geração gráfica da infra-estrutura hierárquica da rede quando deparada com grandes representações hierárquicas se mostra confusa e de difícil contextualização quando se deseja entender os relacionamentos entre os nós. [12].

Quando um determinado usuário ou um grupo de usuários encontra dificuldades para realizar uma tarefa com a interface, dizemos que esta interface apresenta um problema de usabilidade. Essas dificuldades podem ter origens variadas e ocasionar perda de dados, diminuição da produtividade e até mesmo a total rejeição do software por parte dos usuários [13]. A geração gráfica atual do Nagios demonstra problemas típicos de usabilidade de sites *web* e de visualização de grandes estruturas hierárquicas.

Quando a rede gerenciada pelo software começa a crescer a visualização gráfica de sua estrutura fica muito prejudicada. O problema reside no elevado número de informações hierarquizadas que necessitam serem visualizadas ao mesmo tempo com o objetivo de se preservar a compreensão do contexto da informação. A necessidade de utilização de barras de rolagem para visualização dos nós mais distantes do nó raiz faz com que o usuário tenha dificuldade de acompanhar as dependências de cada nó. A perda de contexto prejudica o processo de detecção de erros visto que o torna mais lento e trabalhoso.

Outro problema observado é a sobreposição de alguns nós quando muitos dados são exibidos na tela simultaneamente, tornando confusa a compreensão da estrutura. Além disso, não existe uma ferramenta de busca na estrutura gráfica gerada, sendo assim o administrador da rede tem que ter prévio conhecimento da rede para localização dos nós.

Existem várias abordagens para o desenvolvimento de técnicas de visualização de informações utilizadas para resolver o problema de visualização de grandes hierarquias. Entre elas podemos citar as técnicas que utilizam a abordagem de Foco+Contexto e técnicas que utilizam uma abordagem de Visão Geral+Detalhe [12].

As técnicas de Foco+Contexto fornecem uma representação visual inteira do espaço de informação bem como uma visão detalhada do item selecionado; e Visão-Geral + Detalhe que tem como característica exibir essas duas partes em áreas separadas.

Uma técnica de Foco + Contexto chamada Árvore Hiperbólica vem sendo muito utilizada na *web* para a geração de organogramas, mapas de site e outras visualizações de estruturas hierárquicas. Nesta técnica a expansão e poda dos nós na estrutura são operações que mantêm sempre uma subárvore visível reduzindo, para o usuário a sensação de perda de contexto. Assim, as árvores hiperbólicas são uma representação dinâmica da estrutura hierárquica e representam uma maneira eficiente de exibir árvores complexas com exatidão [12].

A motivação desse trabalho está em utilizar uma Árvore Hiperbólica para a construção de um módulo de monitoramento para o software Nagios aproveitando assim os recursos para visualização de grandes estruturas hierárquicas dessa técnica.

O objetivo principal deste trabalho é o desenvolvimento de um módulo para o Nagios que utilize uma Árvore Hiperbólica para realizar a geração da estrutura gráfica da rede monitorada, visando dessa forma corrigir os problemas de usabilidade do módulo de monitoramento atual do programa. A implantação desse novo módulo no Nagios busca a melhoria nos processos gerenciamento dos serviços de rede.

No capítulo 2 serão mostrados conceitos importantes sobre o gerenciamento de redes de computadores e uma descrição das principais funcionalidades do software Nagios de gerenciamento de redes.

As técnicas de visualização de estruturas hierárquicas serão apresentadas no capítulo 3 juntamente com conceitos sobre usabilidade de software. Serão abordados detalhes sobre a técnica de visualização Árvore Hiperbólica que foi utilizada para o desenvolvimento do módulo para o software Nagios.

No capítulo 4 serão descritos o módulo implementado para o software Nagios, as tecnologias utilizadas para o desenvolvimento desse módulo e os procedimentos desse desenvolvimento.

Os resultados obtidos com a implementação do módulo desenvolvido neste trabalho serão apresentados no capítulo 5. Estes resultados serão avaliados comparando os módulos padrão do software Nagios com o módulo desenvolvido neste trabalho.

No capítulo 6 serão mostradas as conclusões do trabalho com a implantação deste módulo no software de gerenciamento de redes Nagios. Além disso serão relatos os recursos adicionais que serão adicionado ao software em trabalhos futuros.

2. GERENCIAMENTO DE REDES

Rede de computadores refere a um conjunto de computadores autônomos interconectados. Para que um computador seja autônomo é necessário que este possa realizar tarefas básicas sem depender de outro computador, não existindo o conceito de mestre/escravo [14].

As redes de computadores são diferentes de sistemas distribuídos. Em ambos os casos existem vários computadores autônomos interligados, a diferença básica é que enquanto em uma rede de computadores o usuário deve explicitar em qual máquina ele deseja conectar, em um sistema distribuído um software é que automaticamente aloca o processamento das tarefas para o respectivo destino [15].

O gerenciamento de uma rede de comunicação visa um controle mais preciso e eficaz dos equipamentos e serviços que compõem a rede. Para isso é necessário reunir o maior número de informações possíveis ao mesmo tempo de todos os objetos que compõem uma determinada rede [1].

O gerenciamento de redes é uma atividade complexa tendo em vista o crescimento acelerado nas redes de comunicação e o surgimento de novos fabricantes de componentes. O gerenciamento e equipamentos são muitas vezes fornecidos por software do próprio fabricante, contudo em uma organização onde o número de equipamentos ligados em rede é consideravelmente grande, e os equipamentos de rede são diversificados, certamente haverá dificuldade no gerenciamento. Para contornar essa situação temos a idéia do chamado Gerenciamento Integrado, que em suma visa utilizar apenas uma ferramenta de gerenciamento para extrair informações de toda rede de comunicação [1].

Um sistema de Gerenciamento Integrado deve abranger [16]:

- Tratamento de erros;
- Fácil mudança de rota de um determinado tráfego;
- Banco de dados com informações de usuários e objetos rede;
- Opções de acesso remoto ao banco de dados do sistema de Gerenciamento Integrado;
- Recebimentos de avisos por e-mail ou celular;
- Mapas de rede, gráficos estatísticos e relatórios apresentados na tela de forma a facilitar o entendimento do administrador de redes.

No próxima seção serão mostrados conceitos sobre arquitetura de gerenciamento de redes.

2.1. Arquitetura de Gerenciamento de Redes

O conjunto de ferramentas utilizadas para monitorar e controlar recursos e serviços contidos na rede constitui uma arquitetura para um sistema de gerência. Tal arquitetura é composta por quatro camadas de funcionalidades e uma camada de dados, além do protocolo de gerenciamento [17]. A figura 1 [19] mostra as camadas dessa arquitetura.

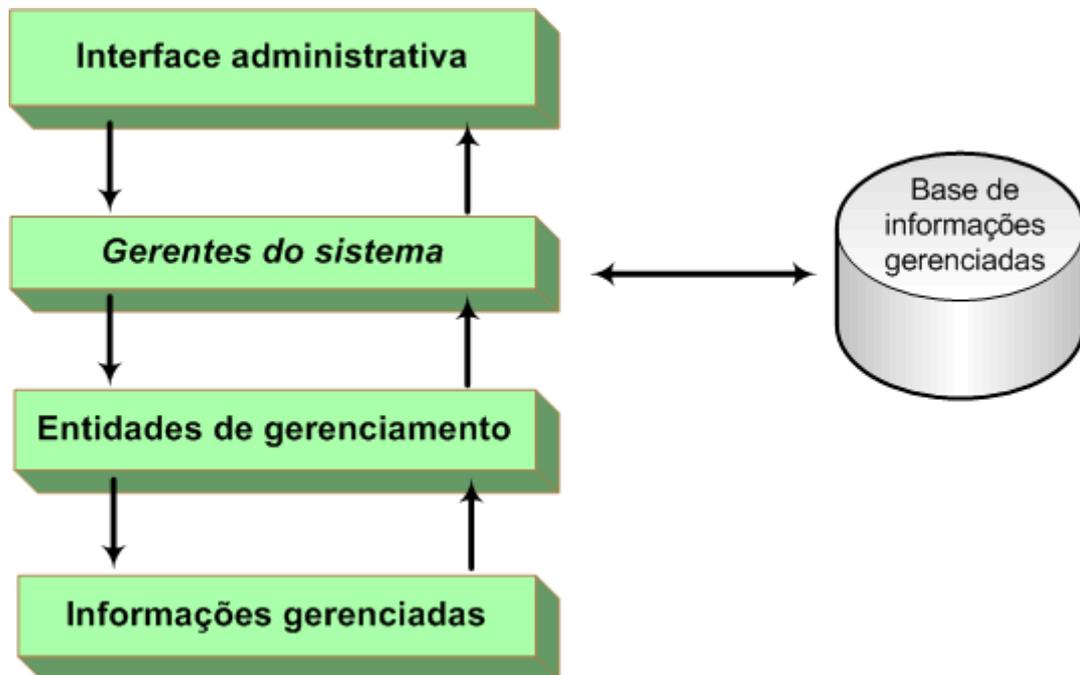


Figura 1 - Camadas de uma arquitetura de gerenciamento.

As camadas da arquitetura de gerenciamento são apresentadas a seguir [17]:

- Base de informações gerenciadas – também conhecida com MIB (*Management Information Base*), responsável pela troca de dados com a base de dados gerenciada. Compreende um conjunto de variáveis usadas para representar informações estáticas ou dinâmicas vinculadas a um determinado recurso ou serviço gerenciado.
- Objetos gerenciados – são dispositivos, aplicações ou quaisquer outros recursos e serviços a serem monitorados e controlados pela arquitetura.

- Entidades de gerenciamento – também conhecidas como agentes são aplicações contidas nas estações do sistema com o objetivo de coletar e armazenar dados acerca dos objetos locais gerenciados, além de responder às requisições feitas pelos gerentes do sistema.
- Gerentes do sistema – uma ou mais entidades de gerenciamento designadas para reunir informações provenientes de outras entidades, formando uma base de dados para análise e geração de relatórios gerais. Adicionalmente, podem programar outras entidades de gerenciamento (agentes) para enviarem notificações de modo automático mediante regras pré-estabelecidas.
- Interface administrativa – interface pela qual o administrador controla e monitora o sistema, configurando, requisitando e recebendo informações ou parâmetros específicos.

O modelo centralizado é o mais utilizado para o desenvolvimento de arquitetura de gerenciamento de redes, esse modelo é composto por um gerente e vários agentes. A comunicação do gerente com os agentes, é realizada por meio de um protocolo de gerenciamento e todas as informações relacionadas ao sistema são armazenadas em uma base de dados central, como mostra a figura 2 [17].

Outro modelo utilizado é o distribuído de gerenciamento, que tem como principal característica a existência de vários subsistemas de gerenciamento independentes entre si, figura 3 [17]. Nesse tipo de arquitetura são criadas entidades de gerenciamento híbridas que ora atuam como gerente e ora atuam como agente. Para isso distribui-se o processamento do gerente do sistema dividindo a arquitetura em uma hierarquia de gerentes e agentes [18]

2.2. Protocolos de Gerenciamento

Os protocolos de gerenciamento de redes descrevem um formato para envio de informações de gerenciamento. Um protocolo do nível de aplicação é utilizado para fazer a comunicação entre gerentes e agentes [2].

Para que possa estabelecer uma comunicação entre gerente e agente, é necessário que ambos compartilhem um mesmo esquema conceitual de informações. Para coletar tais informações é necessário que os agentes as disponibilizem para um ou mais gerentes. Tal tarefa pode ser realizada através algumas técnicas como: *pooling* e *event report* (notificações e relato de eventos).

O *pooling* é utilizado para estabelecer a comunicação entre o agente e o cliente em uma interação do tipo requisição e resposta. Essa técnica é utilizada para obter periodicamente informações armazenadas nas MIBs associadas aos agentes pelos gerentes. Já no *event report* os agentes enviam informações aos gerentes, ficando assim os gerentes aguardando a chegada das informações.

Os protocolos mais utilizados no gerenciamento de redes são o SNMP (*Simple Network Management Protocol*) e o CMIP (*Common Management Information Service Over TCP*) [19].

O protocolo SNMP é apresentado como a primeira solução para gerenciamento de redes baseadas, inicialmente, em TCP/IP. Nesse protocolo é adicionado um agente ao hardware. Este agente nada mais é do que um software que ficará encarregado de coletar dados dos dispositivos e armazená-los em uma base de dados. Tal base de dados deve ficar disponível para posteriormente ser acessada nas estações encarregadas de monitorar a rede.

Dentre as características do SNMP destaca-se a capacidade de monitorar e isolar falhas, gerenciar o desempenho e configuração da rede, além de disponibilizar aplicações com menus e alertas sobre o estado dos serviços da rede.

O SNMP é composto por PDUs (*Protocol Data Units*) mensagens trocadas entre elementos gerenciadores (gerentes) e gerenciados (agentes). O número de PDUs do SNMP é reduzido. Seus criadores procuraram simplificar ao máximo o protocolo, limitando as restrições impostas às antigas ferramentas de gerenciamento, pois, na medida em que se aumenta a complexidade do protocolo e as possibilidades de suas mensagens, dificulta-se a criação das ferramentas de gerenciamento devido ao uso de operações complicadas [19].

O protocolo CMIP possui uma quantidade menor de produtos que a implementam quando comparado com o SNMP. Esse protocolo é baseado no modelo OSI (*Open Systems Interconnect*), e possui características mais robustas em relação ao SNMP. Por conta deste fato o protocolo de gerenciamento (CMIP) e os serviços oferecidos (CMIS) também apresentam maiores complexidades. Entre suas vantagens está a quantidade maior de operações que ela disponibiliza como operações de *CREATE* e *DELETE* que permite que objetos sejam criados e eliminados dinamicamente [17].

2.3. Áreas funcionais do gerenciamento de redes

A arquitetura de gerência do modelo OSI é dividida em áreas funcionais bem caracterizadas. Foram definidas cinco áreas funcionais para prover a necessidade do usuário no gerenciamento de suas redes [19]:

- Gerenciamento de Falhas;
- Gerenciamento de Contabilização;
- Gerenciamento de Desempenho;
- Gerenciamento de Configuração;
- Gerenciamento de Segurança.

Comumente encontramos características comuns entre as diversas áreas funcionais, isso acontece porquê embora as funcionalidades de cada área seja diferente, suas funções podem servir de suporte para funções de outras áreas [17].

2.3.1. Gerenciamento de falhas

A gerência de falhas tem a responsabilidade de monitorar os estados dos recursos, da manutenção de cada um dos objetos gerenciados, e pelas decisões que devem ser tomadas para restabelecer as unidades do sistema que venham a dar problemas. Além disso, essa área funcional deve fornecer relatórios com as anomalias ocorridas na rede, coletar continuamente informações sobre o estado dos recursos gerenciados e assegurar o bom funcionamento dos componentes do sistema [17].

As principais funções do Gerenciamento de Falhas, são:

- Procurar e identificar problemas;
- Receber e resolver notificações de detecção de erros;
- Desencadear seqüências de testes de diagnóstico;

- Isolar os pontos de falhas;
- Corrigir os problemas;
- Manter e examinar o histórico de erros.

2.3.2. Gerenciamento de contabilização

A gerência de Contabilidade provê meios para se medir e coletar informações a respeito da utilização dos recursos e serviços de uma rede, para podermos saber qual a taxa de uso destes recursos, para garantir que os dados estejam sempre disponíveis quando forem necessários ao sistema de gerenciamento, ou durante a fase de coleta, ou em qualquer outra fase posterior a esta. Deve existir um padrão para obtenção e para a representação das informações de contabilização, e para permitir a interoperabilidade entre os serviços dos protocolos OSI [17].

São funções do Gerenciamento de Contabilização:

- Estipular tarifas específicas para utilização os recursos do sistema;
- Registrar níveis de utilização por usuários ou grupos de usuários;
- Gerar relatórios de utilização para planejamento de possíveis expansões do sistema.

2.3.3. Gerenciamento de desempenho

Na gerência de desempenho temos a possibilidade de avaliar o comportamento dos recursos num ambiente de gerenciamento OSI para verificar se este comportamento é eficiente. Preocupa-se com o desempenho atual da rede, através de parâmetros estatísticos como atrasos, vazão, disponibilidade, e o número de retransmissões realizadas [17].

São considerados como funções do Gerenciamento de Desempenho:

- Monitorar os recursos do sistema;
- Determinar níveis de desempenho aceitável;
- Detectar pontos de desempenho abaixo do esperado;
- Garantir o desempenho adequado;
- Registrar eventos relativos ao desempenho dos recursos.

2.3.4. Gerenciamento de configuração

O objetivo da gerência de configuração é o de permitir a preparação, a iniciação, a partida, a operação contínua, e a posterior suspensão dos serviços de interconexão entre os sistemas abertos. Desta forma, tem a função de manutenção e monitoração da estrutura física e lógica de uma rede, incluindo a verificação da existência dos componentes, e a verificação da interconectividade entre estes componentes [17].

Esse gerenciamento possui as seguintes funções:

- Gerenciar os parâmetros relativos a serviços e recursos contidos na rede;
- Preparar, iniciar e encerrar a execução de serviços e recursos gerenciados; e
- Prover meios de modificar configurações em tempo de execução.

2.3.5. Gerenciamento de segurança

O objetivo do gerenciamento de segurança é o de fornecer subsídios à aplicação de políticas de segurança, que são os aspectos observados para que uma rede baseada no modelo OSI seja operada corretamente, protegendo os objetos gerenciados e o sistema, de acessos indevidos de intrusos. Deve providenciar um alarme ao gerente da rede sempre que se detectarem eventos relativos à segurança do sistema [17].

Dentre as funções do Gerenciamento de Segurança, estão:

- Controlar cadastros de usuários;
- Autenticar usuários;
- Autorizar (ou não) o acesso a recursos do sistema;
- Assegurar a integridade, a confidencialidade e a autenticidade das informações;
- Utilizar protocolos de comunicação segura e *firewalls*;
- Registrar eventos do sistema.

No próxima seção serão descritas as principais funcionalidades software Nagios que trata de um software que engloba uma solução centralizada na área de monitoramento de redes.

2.4. Software de monitoramento integrado

Nagios

O Nagios é uma popular aplicação de monitoração de rede de código aberto e licenciado pelo sistema GPL [11]. Ele pode monitorar tanto *hosts* quanto serviços, alertando o analista de TI quando ocorrerem problemas e também quando os problemas forem resolvidos [11].

Foi originalmente criado sob o nome de Netsaint, foi escrito sendo atualmente mantido por Ethan Galstad, junto a um grande número de desenvolvedores que ativamente mantém *plugins* oficiais e não-oficiais.

O monitoramento de serviços como *web* (HTTP, NNTP), Email (SMTP, POP3) são tarefas atendidas pelo Nagios. Esses serviços, em caso de imprevistos, precisam permanecer o menor tempo possível fora de operação, a fim de evitar o comprometimento dos serviços. Desta forma, o Nagios permite o monitoramento dos nós da rede de maneira a perceber ou não a existência de um *host* ou do estado de algum serviço na rede.

Os serviços de verificação no Nagios são exercidos por *plugins* no formato CGI (*Common Gateway Interface*), e podem ser desenvolvidos pelo próprio usuário. Estes CGIs são armazenados em uma única pasta e carregados pelo navegador *web*, quando solicitado.

As principais características do Nagios são [11]:

- Monitoramento de redes e serviços.
- Monitoramento dos recursos de clientes.
- Monitoramento de fatores ambientais.
- Notificação de resultados.
- Definição de hierarquia de redes.

Os computadores “clientes” podem ter seus hardwares monitorados plenamente, com o intuito de obter estatísticas em tempo real da utilização de *hosts* específicos. Dentre os recursos monitorados, pode-se destacar: processos em execução, uso de disco rígido, carga de trabalho do processador e uso de memória RAM [11].

O software Nagios conta também com a geração gráfica de uma árvore hierárquica (Figura 4 Retirada de <http://www.3com.com/osn/assets/nagios1.jpg>) onde o servidor fica no topo, enquanto que os *hosts* seguintes são posicionados ao longo das ramificações. Com essa estrutura é possível diferenciar, por exemplo, clientes desativados de clientes inalcançáveis.

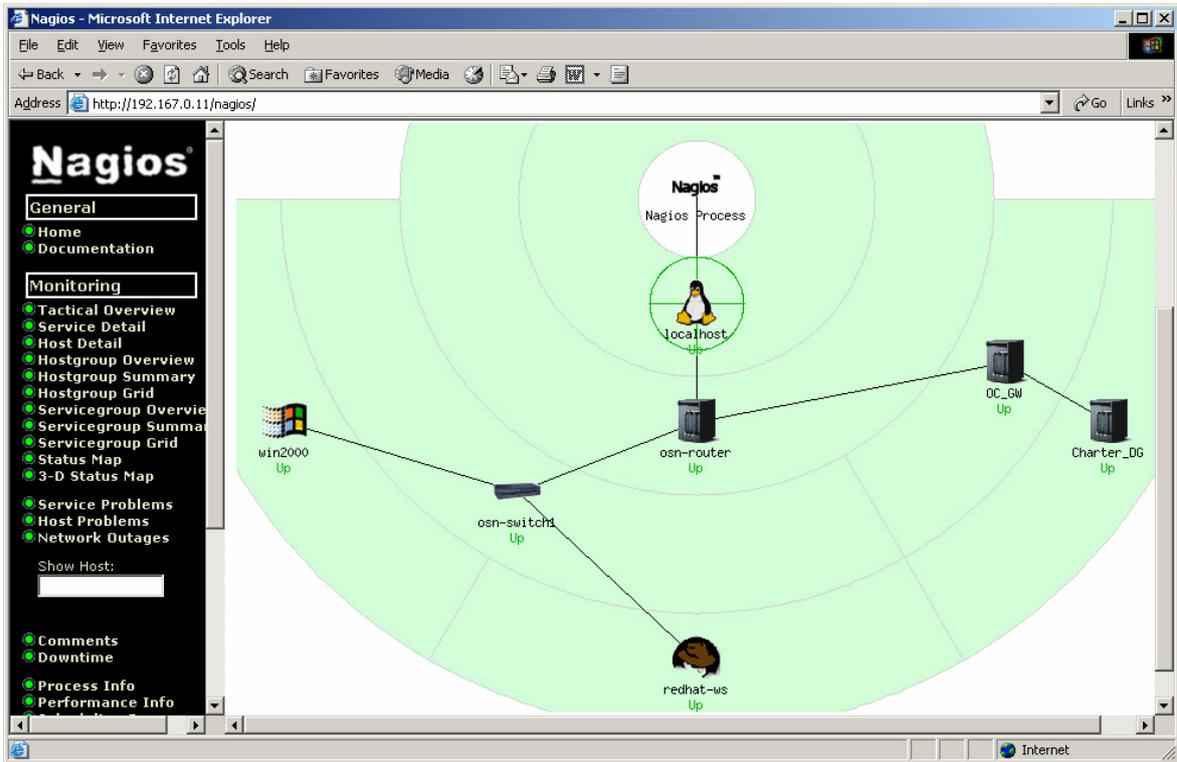


Figura 4 - Geração gráfica da hierarquia de rede.

A apesar de contar com geração gráfica da rede sua representação esta é confusa quando a rede possui muitos nós (equipamentos) de mesma hierarquia e de difícil navegabilidade quando cresce muito em profundidade, sendo necessário a utilização de barras de rolagem para visualizar todos os nós da rede.

Variáveis de ambiente, como a temperatura interna do PC, também podem ser controlados através do Nagios, para isso basta à aquisição do sensor, aparelho disponível no sítio oficial do software, onde também é oferecida uma interface *web* interna que possibilita a alteração dos valores mínimos e máximos pré-estabelecidos. Este equipamento faz a leitura e repassa as informações ao aplicativo, para arquivo em *log*, e pode-se obter o resultado do seu estado através de consultas diretas ao *host* responsável pelo monitoramento [11].

O Nagios tem a capacidade de notificar ao administrador e a informar a um ou mais grupos de contato cadastrados quando um serviço ou equipamento apresenta problemas e quando o problema é resolvido. Essas notificações podem ser realizadas através de

mensagens SMS (Figura 5 <http://nagios.sourceforge.net>), Pager, e-mail ou outros métodos definidos [11].

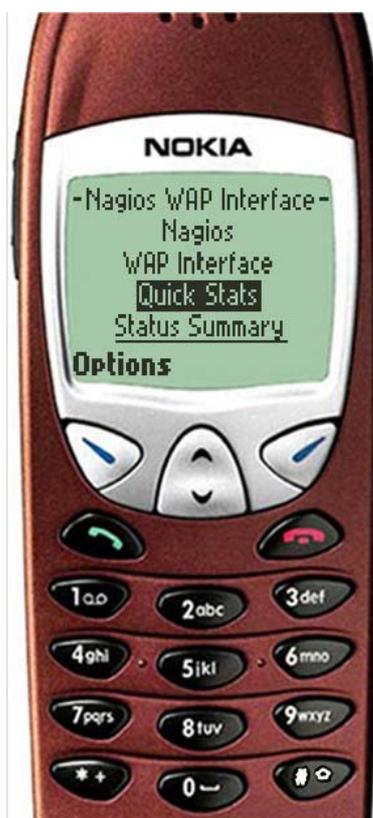


Figura 5 - Interface WAP.

No próximo capítulo são definidos conceitos referentes à usabilidade de softwares.

3. VISUALIZAÇÃO DE INFORMAÇÃO E USABILIDADE DE SOFTWARE

Neste capítulo serão apresentados o referencial teórico sobre os conceitos de usabilidade de software e visualização de informações. Serão mostrados as técnicas tradicionais para visualização de estruturas hierárquicas e os problemas mais comuns encontrados nessas técnicas. Duas abordagens para resolução de problema de visualização de grandes hierarquias serão descritas na seção 3.4 e na última seção serão apresentados detalhes sobre a técnica de visualização denominada Arvore Hiperbólica e as ferramentas utilizadas para sua geração.

3.1. Usabilidade de software

Usabilidade é o termo usado para descrever a qualidade da interação dos usuários com uma determinada interface [20]. Esta qualidade está associada aos seguintes princípios [21]:

- Facilidade de aprendizado;
- Facilidade de lembrar como realizar uma tarefa após algum tempo;
- Rapidez no desenvolvimento de tarefas;
- Baixa taxa de erros;
- Satisfação subjetiva do usuário.

Quando um determinado usuário ou um grupo de usuários encontra dificuldades para realizar uma tarefa com a interface, dizemos que esta interface apresenta um problema de usabilidade. Essas dificuldades podem ter origens variadas e ocasionar perda de dados, diminuição da produtividade e até mesmo a total rejeição do software por parte dos usuários [13].

Algumas métricas ou fatores podem ser usados para a determinação de um problema de usabilidade tais como: desempenho do usuário durante a realização de tarefas, satisfação subjetiva do usuário, correspondência com os objetivos do usuário e adequação a padrões [13].

Desempenho do usuário durante a realização de tarefas: é a observação da realização de tarefas realizadas de forma direta ou indireta por usuários, permitindo a verificação das seguintes métricas:

- Conclusão de tarefas (com sucesso, parcialmente concluída, não-concluída): tarefas que não são concluídas ou o são apenas parcialmente, possuem um forte indício de que existe algum problema de usabilidade;
- Tempo de realização da tarefa: mesmo se concluída com sucesso, um tempo excessivamente longo pode indicar um esforço desnecessário sendo exigido do usuário;
- Ocorrência de erros: vários tipos de erros podem ocorrer durante a realização de uma tarefa. Se o erro é causado por uma operação do usuário por exemplo, deve-se investigar se a interface não induz ao erro através de comandos complexos ou ausência de mensagem adequadas. Se o erro é produzido por uma atividade do sistema, deve-se verificar como o usuário é advertido da ocorrência e que suporte é oferecido pela interface para efetuar a recuperação deste erro.

Satisfação subjetiva do usuário: a usabilidade é também uma qualidade subjetiva que compreende a opinião do usuário da interface; se os usuários estão satisfeitos com a interface, o efeito de eventuais problemas é minimizado.

Correspondência com os objetivos do usuário: é independente das tarefas suportadas pelo sistema. Checa se os objetivos do usuário foram alcançados. Esta é uma métrica que pode ser quantitativa ou qualitativa, de acordo com o que é considerado como objetivo final dos usuários.

Adequação a padrões (normas, recomendações ergonômicas, etc.) grande parte do conhecimento sobre usabilidade é organizado na forma de normas e recomendações ergonômicas tais como as definidas pela ISO9241. Tais recomendações descrevem padrões conhecidos de problemas e, em alguns casos, propõem soluções ou alternativas para evitá-los. A aplicação de tais recomendações durante o desenvolvimento da interface pode realmente evitar ou reduzir vários problemas de usabilidade. Pode-se verificar a usabilidade inspecionando uma interface em relação a tais recomendações. Se a interface as segue pode-se estimar que os problemas de usabilidade foram evitados.

As próximas seções irão tratar alguns detalhes sobre Visualização de Informações, Visualização de estruturas hierárquicas e apresentação de técnica de Contexto Geral + Detalhe e Foco + Contexto.

3.2. Introdução à visualização de informações

A visualização de Informação consiste em utilizar recursos da computação gráfica para auxiliar a visualização, análise e compreensão de um determinado conjunto de dados. Tal auxílio é proporcionado em grande parte pela utilização de forma interativa das formas visuais permitindo que o usuário manipule a forma como os dados são visualizados [22].

A Visualização de Informações estuda formas de transformar dados abstratos em imagens reais ou mentalmente visíveis, de forma a facilitar a sua compreensão e ajudar na descoberta de novas informações contidas nesses dados. Esse processo visa auxiliar no entendimento de um assunto, no qual, sem uma visualização, seria mais difícil de ser compreendido [23].

Utilizando técnicas de visualização de informações, por meio de recursos computacionais, é possível converter dados e apresentá-los visualmente ao usuário em forma de imagens ou outros estímulos sensoriais, de forma que possam ser mais bem compreendidos. Essas representações podem ser distribuídas em três classes: unidimensional, bidimensional ou tridimensional, que são definidas de acordo com a dimensão do espaço onde os elementos geométricos utilizados estejam situados [22].

Podemos considerar a visualização de informação como sendo o processo de mapeamento de dados para a forma visual como mostra a figura 6 [24].

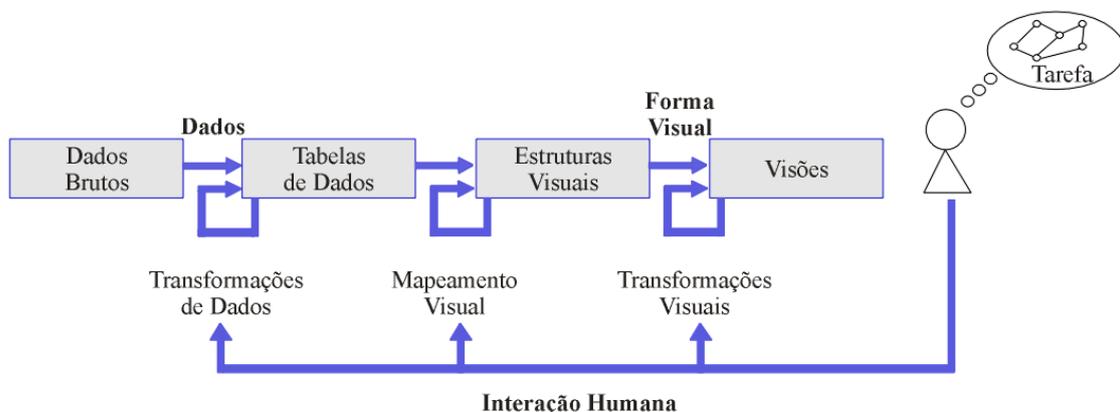


Figura 6 - Modelo de Referência para Visualização.

As tabelas constituem uma das maneiras mais simples de organizamos conjuntos de dados. Nelas as linhas correspondem a um registro e as colunas representam o número de atributos que determinam a dimensionalidade dos dados. No entanto, os seres humanos possuem dificuldades em memorizar e processar dados tabulares. Já com cenas visuais nota-se uma maior facilidade de processamento das informações apresentadas. Isso pode ser observado, por exemplo, quando são analisados dados dispostos em gráficos, o que torna a interpretação destes dados mais fácil e rápida do que se colocada em formato tabular [24].

3.3. Visualização de estruturas hierárquicas

Os tradicionais métodos de recuperação das informações são associados às entidades que compõem estruturas organizadas hierarquicamente. São exemplos, árvores genealógicas, organogramas, e mapas de *web* sites, que necessitam de mecanismos de exploração a fim de que possam perceber de forma mais intuitiva os relacionamentos existentes entre seus componentes [12].

Os *leiautes* mais utilizados para apresentação de estruturas hierárquicas são:

- Digrama *top-down*;
- Listas delimitadas;
- Listas endentadas;
- Conjuntos aninhados;

No caso do diagrama *top-down* (Figura 7 [12]) a relação entre a largura e profundidade da hierarquia pode trazer dificuldades na visualização. Na medida que árvore cresce linearmente em profundidade, o espaço exigido para a exibição do nível mais baixo cresce de forma exponencial.

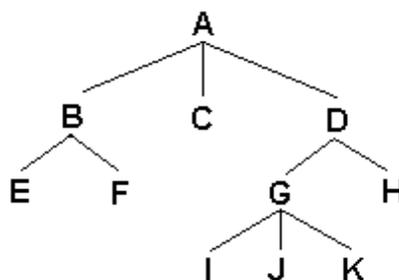


Figura 7 - Digrama *top-down*.

A técnica com listas delimitadas (Figura 8 [12]) têm a vantagem de exibir os relacionamentos pai/filhos de forma mais clara, porém essa técnica faz com que o usuário tenha que percorrer o caminho de todos os nós para construir o modelo mental da hierarquia. Nas listas delimitadas são listados todos os caminhos da raiz a cada nó existente na hierarquia, separando cada elemento com uma barra. Para a percepção global em grandes hierarquias essa forma de representação torna-se inaceitável.

A
A/B
A/B/E
A/B/F
A/C
A/D
A/D/G

Figura 8 - Listas delimitadas.

A lista endentada (Figura 9 [12]) resolve algumas dificuldades inerentes às listas delimitadas, como a exibição da estrutura da árvore. Isso permite a exibição de um maior número de detalhes sobre os nós visualizados, mas introduz outras dificuldades quando as árvores crescem em tamanho. A exibição de apenas um nó por linha ocasiona uma visualização reduzida do trecho da hierarquia, gerando problemas referente à perda de contexto durante a navegação. Outro fato a ser observado é quantidade de linhas paralelas ao longo da margem esquerda da tela, que podem conduzir a uma confusão sobre os relacionamentos pai/filhos existentes na estrutura. Essa estrutura é muito utilizada em alguns navegadores de arquivos, como, por exemplo, no Windows Explorer da Microsoft que exibe para cada nó uma lista de filhos imediatamente após o seu respectivo pai, ligando-os através de linhas.

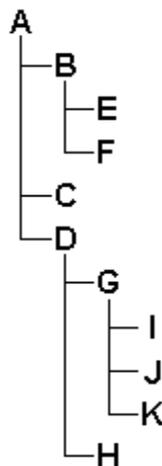


Figura 9 - Listas endentadas.

Outra técnica utilizada mostra os dados através de conjuntos. Essa representação é denominada Conjuntos Aninhados (Figura 9 [12]). Nela a árvore é exibida como um grupo de conjuntos, onde cada nó é desenhado como um subconjunto de seu pai. Embora a área para exibição de informações dentro dos nós seja limitada, a vantagem dessa representação está no melhor aproveitamento do espaço da tela.

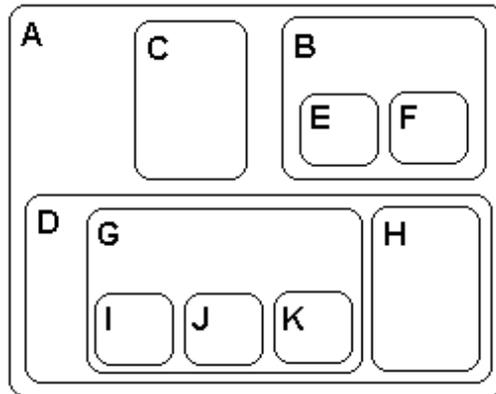


Figura 10- Conjuntos aninhados.

3.4. Visualização de informações de contexto e detalhe

Podemos citar duas abordagens para o tratamento da exibição de estruturas hierárquicas, são elas: Foco + Contexto que fornece uma representação visual inteira do espaço de informação bem como uma visão detalhada do item selecionado; e Visão-Geral + Detalhe que tem como característica exibir essas duas partes em áreas separadas.

3.4.1. Abordagem Visão Geral + Detalhe

A abordagem Visão Geral+Detalhe consiste em separar a área visual da estrutura a ser observada em uma área principal, que é foco do trabalho realizado, e em outras pequenas áreas. A área principal é mais detalhada focando a parte que está sendo concentrado o seu trabalho e as demais são simplificadas. A visão principal é chamada de detalhe, as demais formam a visão geral. [12].

Esse tipo de abordagem envolve cuidadoso balanceamento na distribuição do uso do espaço. Utiliza-se nessa abordagem o processo de *zoom* (Colocar em destaque uma

determinada área visual da tela através do aumento da região que se deseja focar) para obtenção da visão da parte da estrutura que se quer ver em detalhes pode ser conseguido através da utilização de um *zoom* espacial ou *zoom* semântico. No *zoom* espacial a visão de detalhe é apenas uma versão ampliada do trecho da visão geral que interessa. Já no *zoom* semântico, o conteúdo mantém-se o mesmo, mas a aparência é alterada.

Essa abordagem possui uma relação espacial indireta entre a janela de visão geral e a de detalhe. A falta de uma relação mais orgânica e menos arbitrária no posicionamento das vistas pode sobrecarregar a memória além de aumentar o tempo da busca visual, o que pode ser considerado o principal problema dessa abordagem.

Um exemplo prático dessa abordagem é o navegador Microsoft Explorer (Figura 11 [12]). Nele vemos uma primeira parte localizada à esquerda que tem por objetivo exibir uma visão geral da hierarquia, onde cada nó é representado por um ícone que denota o seu tipo, seguido por um texto que o identifica. Os nós que representam arquivos não são exibidos. Eles são posicionados um abaixo do outro, e cada nível é representado a uma certa distância da margem esquerda. Quando o número de nós a serem exibidos excede as dimensões da janela, são exibidas barras de rolagem, tanto no sentido vertical como no horizontal, para permitir o controle da visualização dos elementos da hierarquia. Nesta situação apenas um trecho da visão geral é exibido.

A janela localizada à direita possui como objetivo a apresentação de todos os filhos do nó selecionado na janela de contexto, sem a restrição do tipo de nó. São fornecidas diversas formas de exibição das informações ligadas a esses nós, que podem ser configuradas pelo usuário. Elas podem variar desde uma representação simplificada, com apenas a relação dos ícones e o texto que identifica cada nó, até uma listagem com os principais atributos de cada nó. Cabe destacar que somente os nós existentes no nível selecionado são exibidos. Se existir uma sub-árvore cuja raiz pertença a este grupo, seus descendentes não serão exibidos. Caso o número de elementos visuais ocupe uma área maior que a oferecida pela janela, barras de rolagem são anexadas para permitir a navegação.

Outro exemplo é a utilização dessa abordagem para representar interfaces de visualização de mapas (Figura 12 [12]). Na parte superior direita o usuário dispõe de uma janela de visão geral que permite visualizar todo o mapa. A área acinzentada de formato retangular dentro da janela de visão geral representa a parte do mapa que está sendo mostrada na janela de detalhe. Ainda é possível redimensionar essa área através da manipulação do canto direito inferior que está sombreado. Com isso, o retângulo pode ficar maior ou menor, produzindo os respectivos efeitos de *zoom in* e *zoom out* na janela de detalhe.

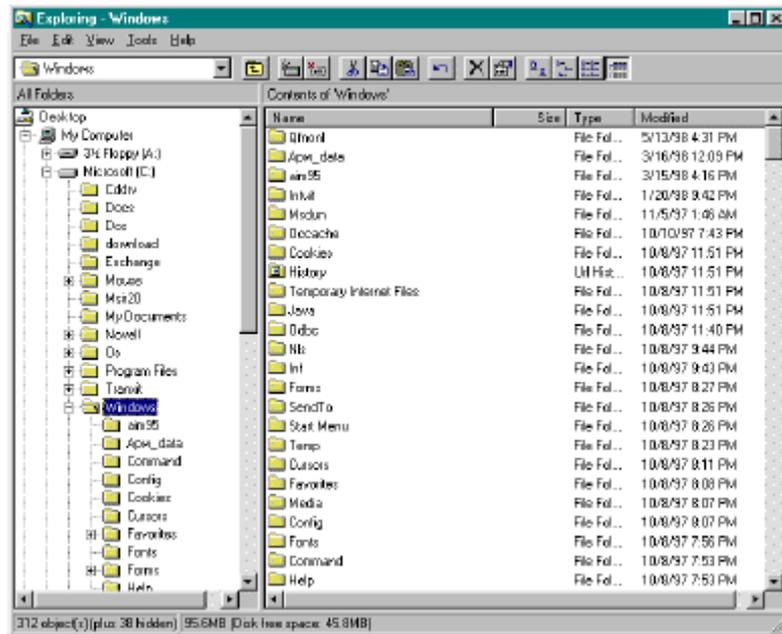


Figura 11- Microsoft Explorer Browser.

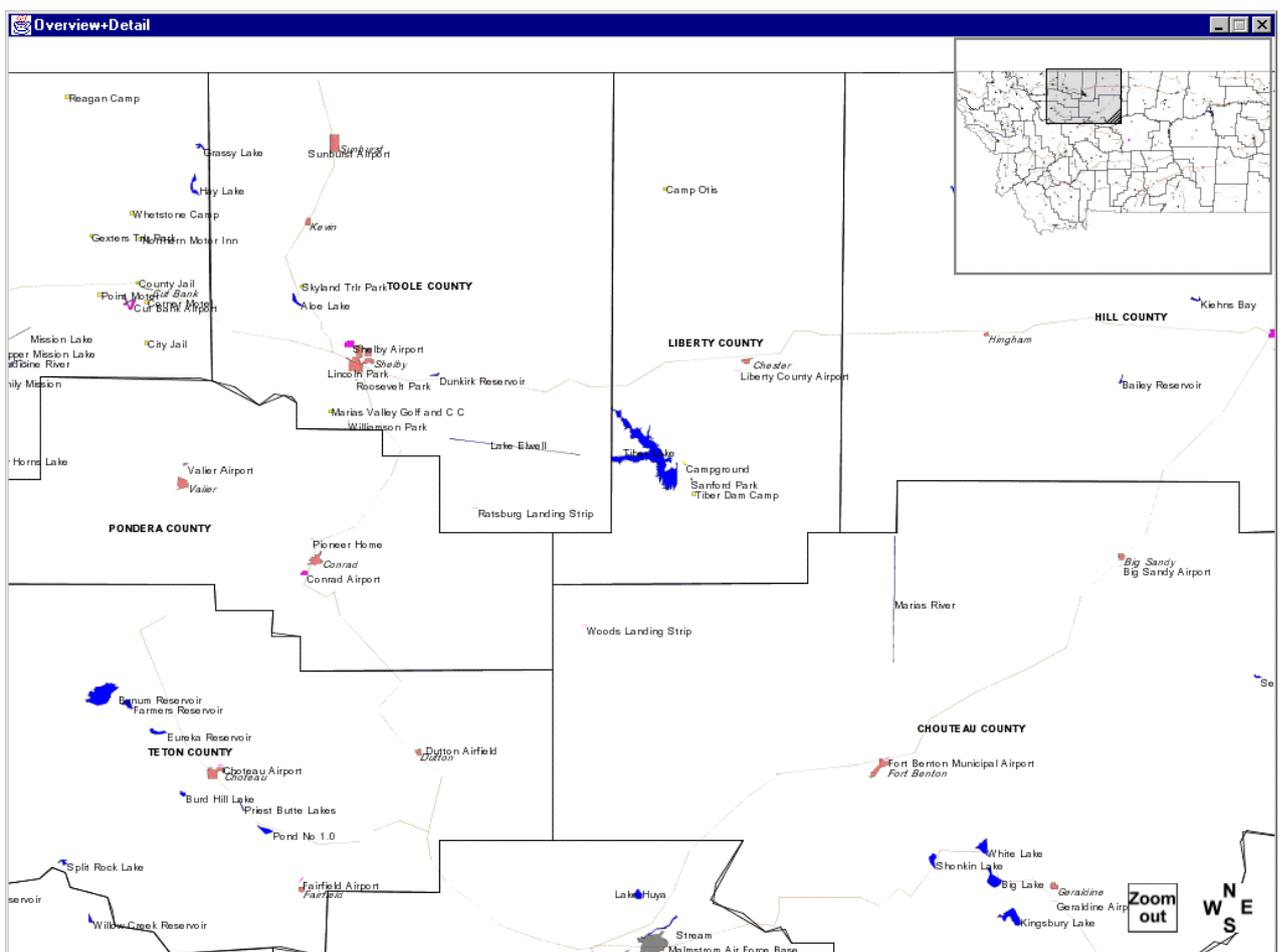


Figura 12- Interface Visão Geral+Detalhe utilizada na exibição de mapas.

3.4.2. Abordagem Foco + Contexto

Quando trabalhamos com grande volume de informações pode ocorrer que alguns elementos sobreponham a outros. A desordem visual é decorrente da dificuldade de reconhecimento e interpretação (pelo usuário) de muitos elementos (não necessariamente sobrepostos) presentes na visualização o que força uma sobrecarga no seu sistema cognitivo [22].

Para resolver estes problemas algumas técnicas utilizam características de realismo, tais como sombra e transparência permitindo ao usuário a localização mais clara dos objetos. Podem ser fornecidos mecanismos básicos de manipulação geométrica como rotação, mudança de escala e translação dos objetos para a minimização desses dois problemas. Algumas técnicas podem apresentar o problema de desorientação visual, que ocorre quando o usuário tem dificuldade na manutenção da atenção por ocasião de troca do ponto de vista e retorno a situações anteriores. Técnicas de foco+contexto tentam minimizar esta desorientação evitando que informações desapareçam (elas apenas tem dimensão reduzida) e apresentando transições suaves animadas entre dois momentos no processo interativo.

Nas técnicas de foco+contexto procura-se ampliar a região de interesse, enquanto se compacta o restante da imagem (do contexto). Como essa abordagem emprega uma distorção da imagem original, ela demanda maior tempo para ser compreendida.

As principais variações da técnica de Foco+Contexto são: *Fish-eye* e Navegador Hiperbólico [23].

A técnica de visualização *Fish-eye* produz um efeito semelhante ao de um olho de peixe ou de uma lente de aumento. Ela permite uma visão mais detalhada de uma região de interesse sem haver perda de seus arredores, através de uma taxa maior de ampliação no centro da região de interesse e decrescente no sentido da periferia da imagem [25].

Na visualização *Fish-eye* os elementos que estão localizados próximos ao local que contém o ponto de interesse do usuário, denominado foco, são exibidos com mais detalhes, que vão diminuindo à medida que se afastam desse ponto. Esse efeito é conseguido através da aplicação de uma função denominada DOI (*Degree-Of-Interest*), que tem por objetivo obter o grau de interesse de um determinado componente fornecendo o nível de detalhe de sua exibição. A função leva em conta a importância do nó na estrutura e sua respectiva distância em relação ao foco que é determinado pelo usuário. Dessa forma ela é definida como:

DOI = API – D, no qual API (*A Priori Importance*) representa o grau de relevância do elemento na estrutura e D (*Distance*) a distância dele em relação ao foco.

O Navegador Hiperbólico, representado pela figura 13 [23], combina Foco+Contexto com desenho radial de árvores para auxiliar na exploração de grandes hierarquias. Nessa representação, mudanças de foco podem ser realizadas através de movimentos simples do mouse objetivando selecionar vértices de maior interesse. Isso move o vértice selecionado para o centro da tela, ocasionando uma compactação das informações que estão distantes do mesmo. Além disso, o sistema realiza transições gradativas e suaves da mudança do ponto de foco, a fim de preservar o a o modelo geral construído para aquela estrutura [26].

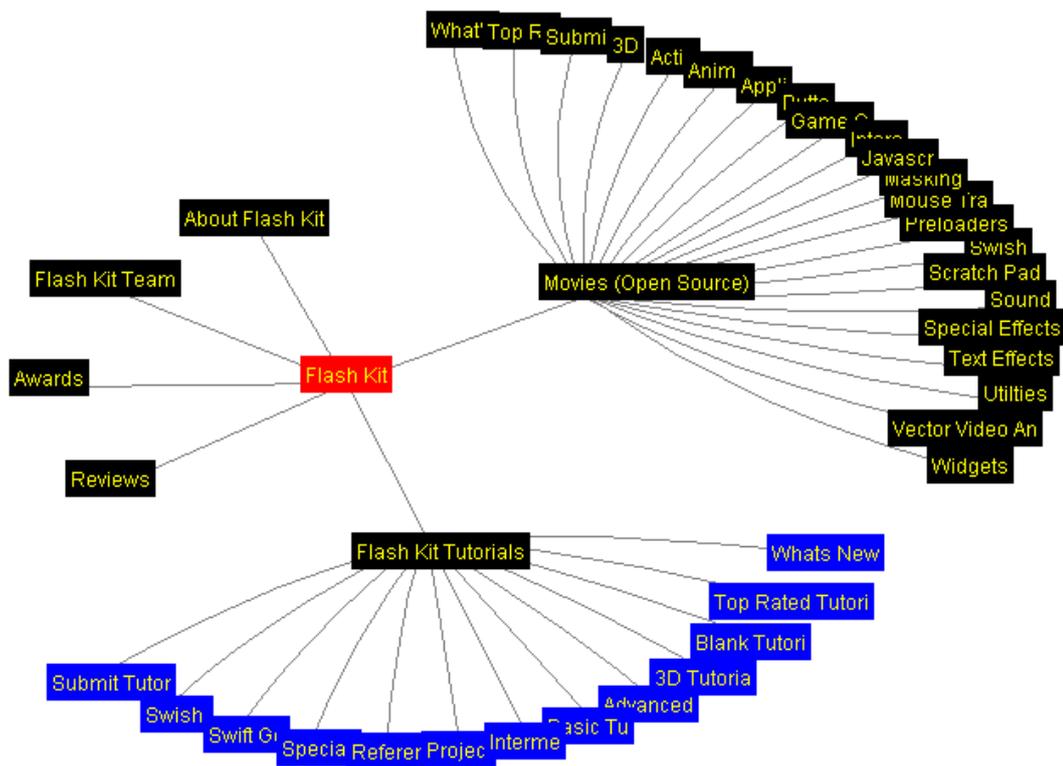


Figura 13 - Navegador Hiperbólico.

3.5. Árvore Hiperbólica, HiperNavegador e HiperEditor

Árvore Hiperbólica é um recurso de visualização gráfico, que visa facilitar a navegação em conjuntos de dados hierárquicos muito grandes. Segue a técnica, Navegador Hiperbólico, uma das técnicas de visualização baseado no foco + contexto, discutido na seção anterior. Nessa representação utiliza-se como base, a forma visual de uma estrutura de árvore em espaço hiperbólico delimitado por uma região circular. O foco da observação é visualizado no centro do círculo, quanto maior a distância do centro maior é o número de informações apresentadas, de modo que o foco e seu contexto imediato, são de fácil legibilidade e as regiões próximas à borda do círculo apresentam uma grande densidade de informações [26].

Esta técnica é bastante utilizada em mapas de sites e como ferramenta de navegação em diversas hierarquias. Nela pode ser atribuído aos nós *links* para outras páginas, além de poder serem personalizadas as cores dos nós [27].

A o setor de informática da Embrapa (Empresa Brasileira de Pesquisa Agropecuária) desenvolveu ferramentas que incorporam algoritmos para apresentação de hierarquias em espaço hiperbólico. O objetivo dessas ferramentas é utilizar esta técnica de visualização de informações para que essa estrutura pudesse ser utilizada em páginas da *web*. Com isso foram criados dois softwares livres, o HiperNavegador e o HiperEditor. A primeira ferramenta permite a visualização de uma árvore hiperbólica em páginas *web*, a segunda permite a edição da estrutura da árvore hiperbólica que será apresentada pelo HiperNavegador [28].

Ambas as ferramentas utilizaram como base a biblioteca JAVA para visualização da geometria hiperbólica implementada por Bouthier [29].

3.5.1. HiperNavegador

O HiperNavegador foi construído com o objetivo de fornecer acesso rápido a arquivos remotos armazenados em servidores *web*. Dentre seus principais recursos podemos citar:

- Mecanismo de busca: destaca o caminho percorrido do termo pesquisado até o nó central da árvore.

- Links associados aos nós da árvore.

Para a utilização do HiperNavegador uma página HTML deve ser construída no intuito de invocar uma *applet* (aplicativos que se servem da JVM, *Java Virtual Machine*, existente na máquina cliente ou embutida no próprio navegador do cliente para interpretar o seu *bytecode*) passando os parâmetros necessários para exibição da hierarquia em questão[28].

Para o funcionamento do software basta que o navegador *web* seja capaz de executar uma *applet* JAVA.

3.5.2. HiperEditor

Para a construção da árvore hiperbólica a Embrapa desenvolveu uma ferramenta baseada em formulários *web*, denominada HiperEditor. Essa ferramenta tem como objetivo a construção remota da árvore hiperbólica através do navegador [27].

Com o HiperEditor é possível definir:

- A ligação entre os nós
- O título dos nós
- As cores dos nós.
- Os links dos nós.

Após a árvore hiperbólica ser salva, o HiperEditor gera um arquivo com extensão .htz. O HiperNavegador deverá referenciar esse arquivo para exibição da árvore, seguindo o modelo esquematizado na figura 14 [28].

A utilização do HiperEditor torna flexível a geração da árvore hiperbólica porém trata-se de uma geração manual e estática. Sendo assim o HiperEditor não proporciona uma geração dinâmica da árvore hiperbólica, sendo inviável sua utilização em sistemas onde as mudanças nos nós representados acontecem de maneira freqüente.

O próximo capítulo apresenta a metodologia aplicada no desenvolvimento da árvore hiperbólica como módulo de monitoramento para o software Nagios. Esse módulo tem com principal característica a geração dinâmica da árvore hiperbólica baseado no funcionamento interno de geração da árvore através do HiperEditor.

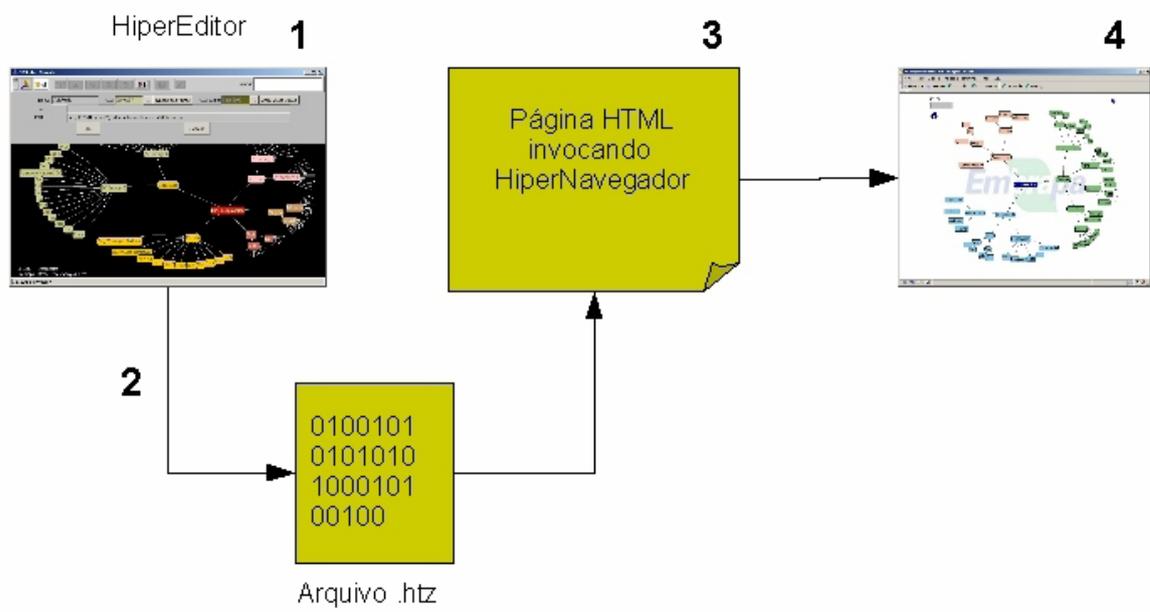


Figura 14 - Esquema HiperEditor x HiperNavegador.

4. METODOLOGIA

O tipo de pesquisa utilizada neste trabalho é a Pesquisa-ação. Este é um tipo de pesquisa social com base empírica que é concebida e realizada em estreita associação com uma ação ou com a resolução de um problema coletivo, no qual os pesquisadores e os participantes representativos da situação ou problema estão envolvidos de modo cooperativo participativo [30].

Na primeira seção deste capítulo serão mostradas as tecnologias utilizadas para a construção do módulo proposto para o software Nagios e descrito o ambiente testado.

Na seção 4.2 será descrito os passo para o desenvolvimento do módulo e suas principais características.

4.1. Materiais

Este trabalho foi desenvolvido utilizando as seguintes tecnologias:

- JAVA.
- JAVA – APPLET.
- PHP.
- HTML.

O ambiente de teste foi à rede da Universidade Federal de Lavras. Para o experimento foi utilizada uma máquina localizada no Centro de Informática da universidade, em um ponto estratégico na rede física da universidade. A configuração da máquina na qual está instalado o Nagios é a seguinte:

- Processador Pentium IV, 1,76 GHz
- 512 MB de memória RAM.
- 80 GB de HD.

Foi utilizado também o HiperNavegador, ferramenta desenvolvida pela Embrapa sobre licença GPL (*General Public License*), responsável pela visualização da estrutura hierárquica através de um navegador.

Para o funcionamento do modulo é necessário ter a máquina virtual JAVA instalada no servidor. Os arquivo do módulo devem ser salvos na pasta *web* do Nagios, juntamente

com os demais arquivos com extensão `.html`. No ambiente testado os diretórios dos arquivos são:

- `usr/local/nagios/etc`: localização do arquivo `host.cfg`, responsável pelo cadastro dos equipamentos gerenciados.
- `usr/local/nagios/var`: localização do arquivo que armazena o estado dos serviços que estão rodando em cada equipamento da rede.
- `usr/local/nagios/share`: pasta onde estão localizados a página do software Nagios. Os arquivos responsáveis pelo funcionamento do módulo criado nesse trabalho deveram ser colocados neste diretório.

O módulo desenvolvido neste trabalho consiste em fazer a geração dinâmica da árvore hiperbólica através da leitura direta dos arquivos de configuração do Nagios. Para realizar tal tarefa o módulo conta com os seguintes arquivos:

- **Biblioteca Java do Software HiperNavegador.**
- **Arquivo `Converte.java`:** classe em linguagem Java adaptada para gerar o arquivo com extensão `.htz` utilizado pelo HiperNavegador na geração da árvore hiperbólica.
- **Arquivo `Converte.class`:** arquivo compilado da classe `Converte.java`.
- **Arquivo `no.php`:** classe construída em linguagem PHP que mapeia as informações de `host` (equipamentos da rede).
- **Arquivo `servico.php`:** classe construída em linguagem PHP que mapeia as informações sobre os serviços de rede que estão rodando em cada `host`.
- **Arquivo `geracsv.php`:** principal arquivo do módulo, seu código fonte foi construído em linguagem PHP. Ele é responsável por traduzir as informações contidas nos arquivos de `log` do Nagios em um arquivo com extensão `.csv` que posteriormente será lido para a geração da árvore hiperbólica pelo HiperNavegador. Este código utiliza as duas classes em linguagem PHP citadas nos itens anteriores para realizar a tradução. Além disso, o código também é responsável por invocar a classe em linguagem Java já compilada `Converte.class`, passando como parâmetro a ela o arquivo com extensão `.csv` por ele gerado. Depois esse invoca o arquivo `navega.html` que utiliza o arquivo `.htz` gerado através da classe `Converte.class` para ativar o HiperNavegador e gerar a árvore hiperbólica.

Na próxima seção serão descritos os métodos utilizados na criação do módulo para o software Nagios.

4.2. Modelo de geração dinâmica da árvore hiperbólica

O modelo para geração da árvore hiperbólica esquematizado na figura 14 do capítulo 4.4.2 mostra a visualização de uma estrutura hierárquica gerada através de software HiperEditor. A idéia do desenvolvimento desse módulo é utilizar o HiperNavegador dentro do Nagios para exibição de sua hierarquia de rede. No entanto, para a geração dessa estrutura através do software Nagios o HiperEditor não pode ser utilizado, já que trata de uma geração estática e manual da hierarquia.

A estrutura gráfica de monitoramento padrão do software Nagios é gerada dinamicamente. Cada mudança no estado de determinado nó é refletida diretamente na hierarquia visualizada. Para realizar tal tarefa, a cada vez que o Nagios consulta o estado de um determinado equipamento na rede, ele escreve as informações coletadas em um arquivo de *log*. Toda vez que é requisitado a visualização da hierarquia de rede, esse arquivo de *log* é consultado, e assim posteriormente é montada a estrutura gráfica da hierarquia.

Além disso, para geração dessa estrutura é necessário que o administrador do Nagios configure um arquivo de *hosts*. Nesse arquivo devem ser cadastrados todos os *hosts* (equipamentos de rede) que serão exibidos, assim como as informações características de cada *host*, tais como: nome do *host*, ip, intervalo de notificação entre outras informações.

Tendo o arquivo de *host* devidamente configurado, a idéia do software desenvolvido neste trabalho é utilizar o mesmo mecanismo, anteriormente citado, para a atualização da árvore hiperbólica que será gerada pelo módulo.

A Embrapa disponibiliza também no pacote de instalação do HiperEditor/HiperNavegador um algoritmo geração do arquivo *.htz*, citado no capítulo anterior, através de um arquivo *.csv* sem a utilização do HiperEditor.

Para essa geração o arquivo *csv* deve estar da seguinte forma:

- Cada linha do arquivo deve ser iniciada pelo nó raiz.

- Cada nó da árvore deve ser separado por ponto e vírgula (;).
- Cada linha representa um caminho da raiz até um determinado nó.

Neste arquivo não são passadas informações como cor do nó, *hint* (Mensagem informativa do nó) e url (link do nó).

A partir dessas informações o desenvolvimento do módulo seguiu os seguintes passos:

- **Modificação do leiaute do arquivo csv:** foi acrescentado no leiaute do arquivo com extensão .csv as seguintes informações: cor, estado (up ou down), *hint* (Descrição dos serviços que estão rodando na máquina e seu estado) e ip. Os nós continuam sendo separadas por ponto e vírgula (;) e as informações adicionais são separados por uma barra vertical (|). Esse arquivo foi chamado de Geral.csv.
- **Criação de código que faz a tradução dos dados dos arquivos do Nagios no leiaute do arquivo .csv modificado no passo anterior:** Este código foi feito na linguagem PHP e utiliza duas classes no.php (contém informações a respeito do arquivo de *hosts* do Nagios) e service.php (contém informações a respeito dos serviços rodando em cada nó). Esse arquivo foi chamado de geracsv.php.
- **Modificação do código que gera o arquivo com extensão .htz através do arquivo com extensão .csv:** como o leiaute do arquivo .csv foi modificado, torna necessária também a modificação desse arquivo. Esse arquivo foi chamado de Converte.class. Os passos anteriormente citados seguem o esquema da figura 15.

Para finalizar, o arquivo que possui as opções de menu do Nagios foi modificado. O arquivo se chama side.html e está localizado na pasta *web* do software. Foi adicionado no arquivo mais uma opção de menu, com o nome *Hiperbolic Tree* e seu link foi modificado para apontar para o arquivo geracsv.php. Esse script PHP além de fazer a tradução dos dados dos arquivos do Nagios no leiaute do arquivo com extensão .csv modificado, executa o programa Converte.class passando como parâmetro o arquivo Geral.csv por ele criado. Esse programa cria o arquivo seg1.htz que é utilizado para gerar o *applet* da árvore hiperbólica. Depois de criado o arquivo seg1.htz, o próprio script PHP chama o arquivo navega.html que é responsável por invocar o applet montado através do arquivo seg1.htz através do HiperNavegador.

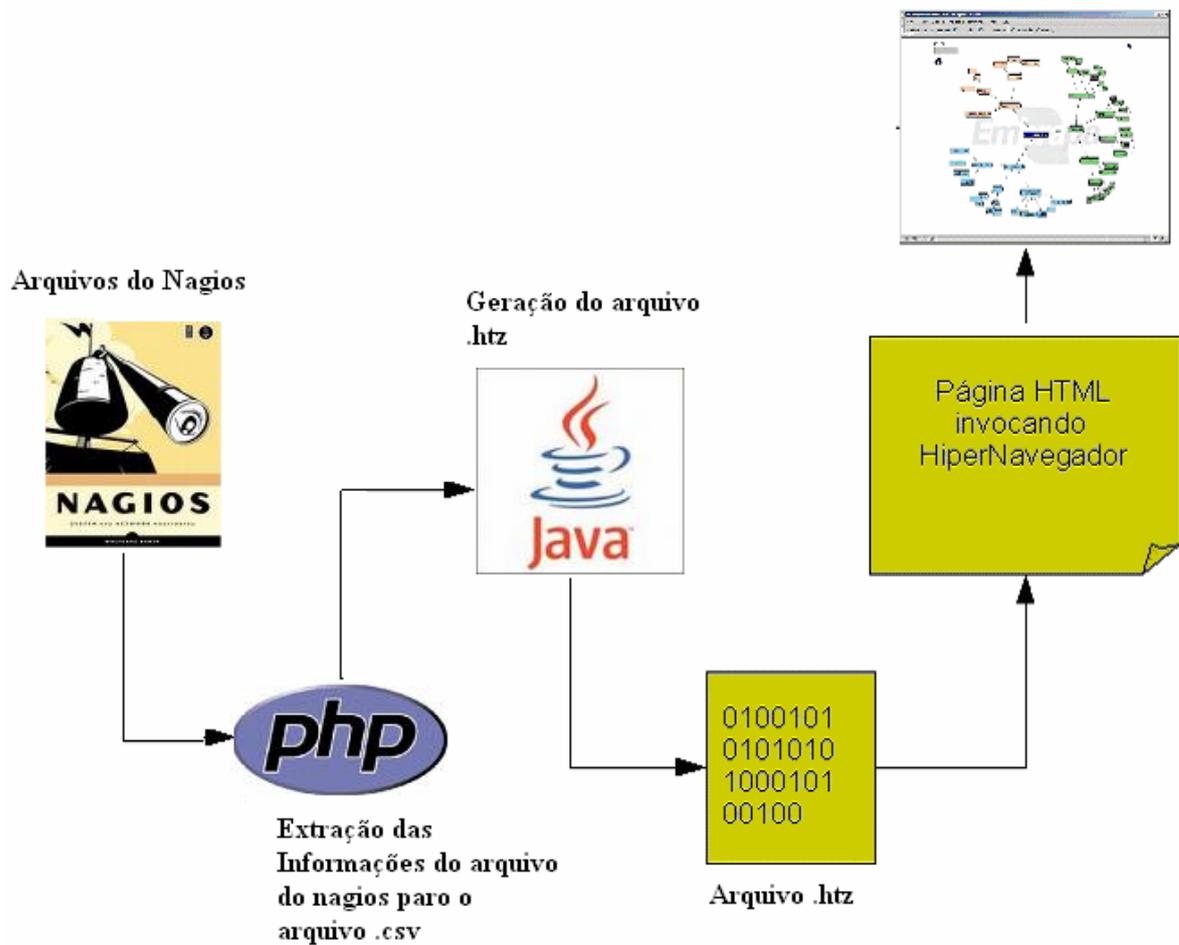


Figura 15 - Navegador Hiperbólico x Nagios

No próximo capítulo serão apresentados os resultados desse projeto com a implantação do módulo desenvolvido no Nagios. Para isso será feito um comparativo entre o modelo de visualização padrão do Nagios com o modelo de visualização implementando com o módulo criado. Serão avaliados os ganhos na eficiência do gerenciamento da rede após a implantação do software.

5. RESULTADOS E DISCUSSÃO

O Nagios conta com dois modelos de geração gráfica da estrutura hierárquica da rede para auxílio na tarefa de monitoramento:

- *Balanced tree*.
- *Circular* e suas variações (*Circular Marked Up*, *Circular Balloon*).

Esses modelos apresentam problemas de usabilidade comuns à interfaces *web* e a visualização de grandes hierarquias. A seguir serão apresentadas as características de cada modelo de visualização.

5.1. Modelo de visualização *balanced tree*

Esse modelo tem como característica a visualização de toda hierarquia no formato de uma árvore balanceada. Cada nó pertencente a um mesmo grau hierárquico, é exibido na mesma linha horizontal. Esse modelo pode ser visualizado em várias escalas e vários tamanhos. Os nós escritos com a palavra *up* estão respondendo, os nós escritos com palavra *down* não estão respondendo aos testes aplicados pelo Nagios. Quando está com um sinal de interrogação significa que o nó não possui imagem representativa do nó. A figura 16 demonstra um exemplo desse modelo de visualização.

Como problemas de usabilidade desse modelo podemos citar:

- **Aparecimento de barras de rolagem:** a necessidade de utilizar barras de rolagem para a visualização de todos os nós da rede torna lento o processo de pesquisa dos nós da rede.
- **Perda de contexto:** como estamos trabalhando com hierarquias a interpretação dos dados está fortemente ligada aos nós e suas relações de parentesco. A utilização de barras de rolagem para chegar aos nós mais distantes da raiz ocasionam uma perda de contexto. Essa perda de contexto proporciona uma maior dificuldade na interpretação dos dados exibidos.
- **Visualização confusa dos nós:** mudanças na escala de visualização reduzem a perda de contexto, contudo torna a visualização dos nós muito confusa, ficando muito difícil à identificação dos nós.

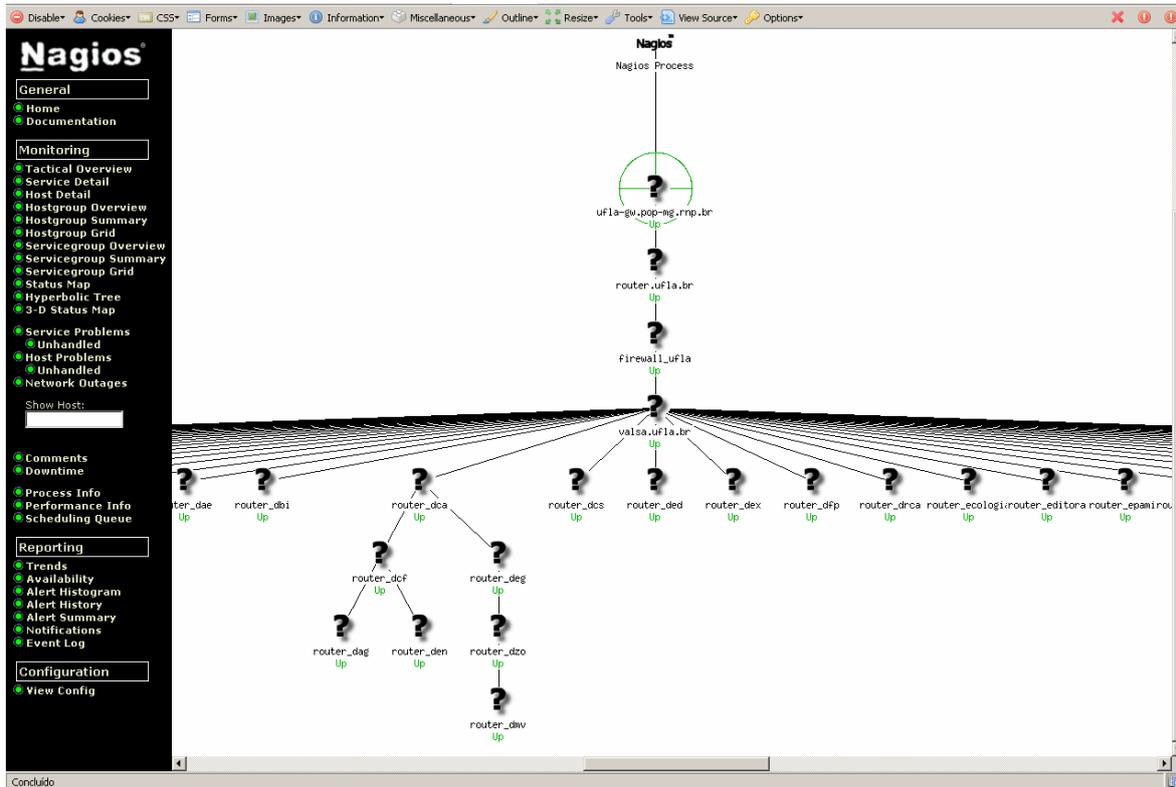


Figura 16 - Modelo de visualização *balanced tree*

5.2. Modelo de visualização circular

Esse modelo tem como característica exibir a estrutura hierárquica em formato circular. Os nós de mesmo grau hierárquico são dispostos em um mesmo círculo. Nós de maior hierarquia são exibidos nos círculos menores, enquanto os nós filhos são exibidos nos círculos mais externos.

A figura 17 demonstra o modelo de visualização *Circular Marked Up* que é uma variação o modelo circular. Neste modelo cada camada hierárquica da rede é delimitada por círculos. Esse também pode modelo pode ser visualizado em várias escalas e vários tamanhos.

Como no modelo anterior os nós escritos com a palavra up estão respondendo, os nós escritos com palavra down não estão respondendo aos testes aplicados pelo Nagios. Quando está com um sinal de interrogação significa que o nó não possui imagem representativa do nó.

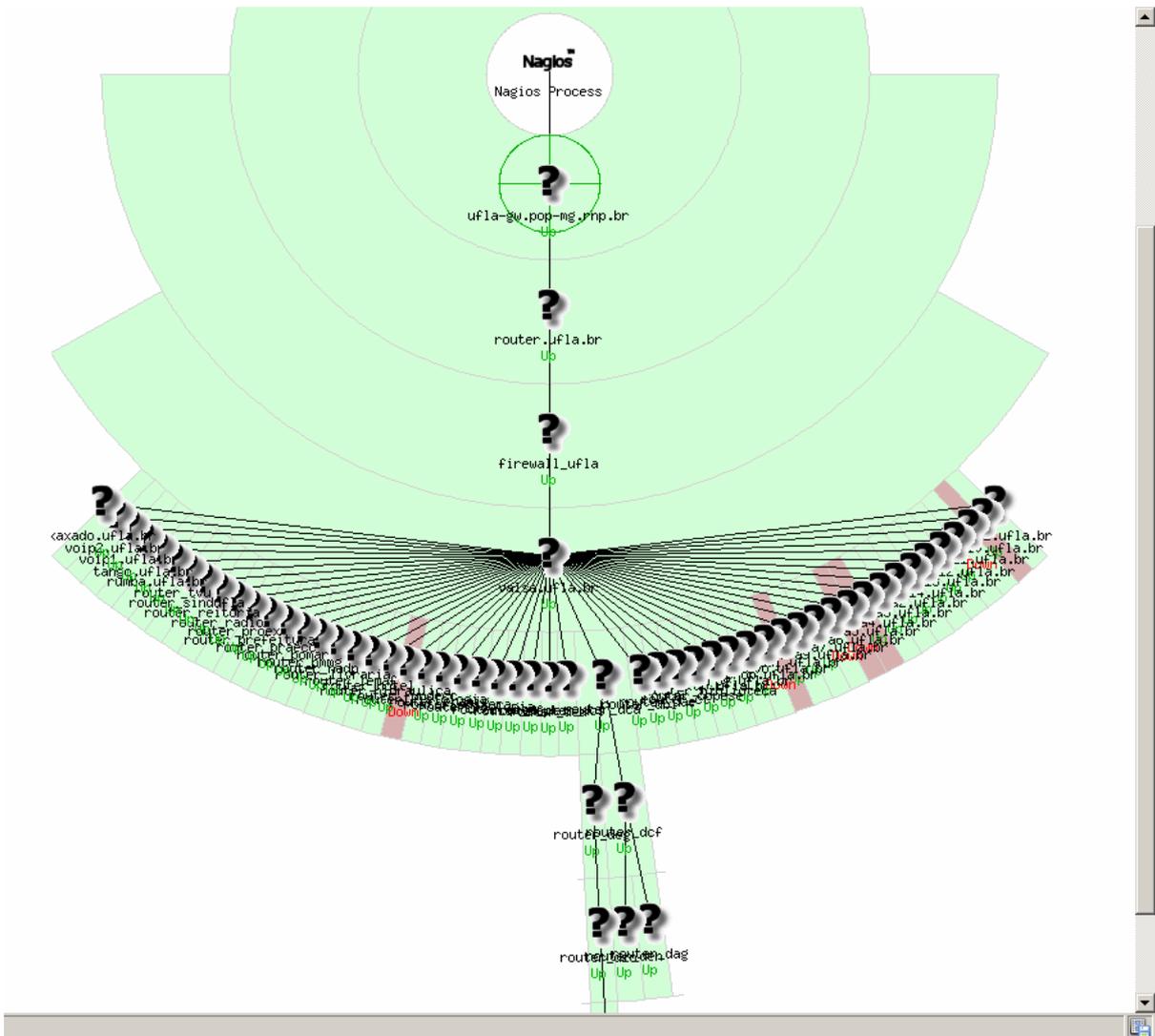


Figura 17 - Modelo de visualização circular *marked up*

Como problemas de usabilidade desse modelo podemos citar:

- **Aparecimento de barras de rolagem:** como no modelo anterior o problema das barras de rolagem também foi evidenciado.
- **Perda de contexto:** a perda de contexto embora diminuída também foi encontrada nessa visualização quando existe um número grande de nós em hierarquias distintas.
- **Visualização confusa dos nós:** nesse modelo mesmo sem a mudança na escala podemos notar que a identificação dos nós está comprometida devida a sobreposição dos nós com seus vizinhos no mesmo grau hierárquico.

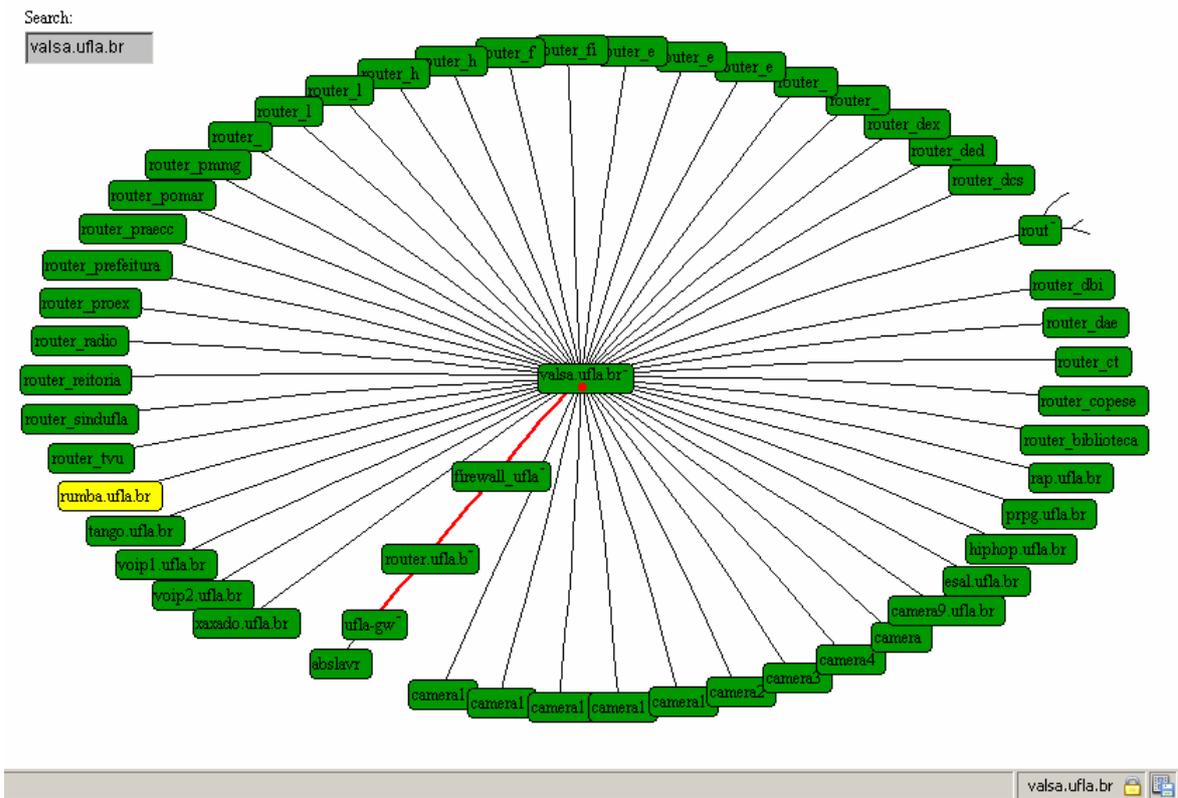


Figura 19 - Visualização com nó valsa.ufla.br no centro.

Nesse modelo os nós de cor verde significam que todos os testes do Nagios estão respondendo para o equipamento daquele nó. Os nós de cor amarela significam que alguns serviços do equipamento daquele nó não estão respondendo aos testes do Nagios. E finalmente os nós representados com cor vermelha não respondem a nenhum teste de Nagios para aquele nó.

5.4. Melhorias na usabilidade do Nagios

A implantação do módulo com o a técnica de visualização com árvore hiperbólica proporcionou as seguintes melhorias na usabilidade do software Nagios:

- **Aparecimento de barras de rolagem:** o aparecimento de barras de rolagem evidenciado nos modelos *balanced tree*, circular e suas variações não é mais problema com o módulo implantado. Para chegar em nós mais distantes da raiz, basta clicar sobre o nó mais próximo que deseja visualizar e arrastá-lo para o centro da tela de forma simples e rápida.

6. CONCLUSÃO

As estruturas de visualização padrão do software Nagios são baseadas em modelos clássicos de visualização de hierarquias, apresentando problemas de visualização na estrutura hierárquica e problemas de usabilidade da ferramenta de monitoramento quando deparados com grandes hierarquias de redes.

A utilização da abordagem Foco + Contexto aplicada na técnica Árvore Hiperbólica para resolução do problema de visualização de grandes representações hierárquicas teve resultados positivos para a representação de estruturas de redes.

Os problemas de usabilidade como perda de contexto provocado pelo aparecimento de barras de rolagem foram significativamente reduzidos com a técnica implementada. As operações de clique e arraste para movimentação em tempo real dos nós tornaram mais rápida a navegabilidade através da visualização da árvore.

O módulo implementado contribuiu também com a construção de um modelo de geração dinâmica da árvore hiperbólica até então inexistente no modelo original de geração da árvore hiperbólica através do HiperEditor.

Além disso, o módulo implementado adicionou ao Nagios um mecanismo de busca através da visualização da hierarquia de rede. Recurso não existente no modelo padrão de monitoramento do software Nagios. Esse mecanismo agilizou o processo de busca por *hosts* e serviços na rede.

O módulo para o software Nagios implantado com esse trabalho melhorou o processo de monitoramento de redes no ambiente testado (Universidade Federal de Lavras) e mostrou resultados promissores, sendo que sua eficiência está na capacidade de monitorar grandes quantidades de equipamentos de rede de forma rápida e simples.

6.1. Trabalhos futuros

Como trabalhos futuros podemos citar as seguintes melhorias:

- Criação de um instalador para o módulo.
- Criação de uma área no Nagios para personalizar as cores dos nós de acordo com cada estado.
- Criação de *links* nos nós para que possa ser realizada alguma ação de gerenciamento do Nagios.

Essas melhorias visam o melhor aproveitamento da capacidade visual humana, melhorar a facilidade de instalação e configuração do módulo e possibilitar a através do módulo criado a correção de alguns erros detectados pela visualização gerada. Para a correção dos erros os *links* adicionados aos nós funcionarão com gatilhos para funções de correções de erros nativas ou não do Nagios.

REFERÊNCIAS BIBLIOGRÁFICAS

- [1] PEREIRA, J. C. **Estudo e Implementação do Nagios para monitoramento de redes locais**, 2007. 78 p. Monografia (Sistemas de Informação do Centro Universitário Jaraguá do Sul) – Centro Universitário Jaraguá do Sul, Jaraguá do Sul.
- [2] PEREIRA, M. C. **Administração e Gerência de Redes de Computadores. Pós-graduação em Ciência da Computação**, 2001 - CPGCC. Universidade Federal de Santa Catarina - UFSC, Florianópolis - SC.
- [3] MRTG. **MRTG Official Web Site**, Disponível em <http://oss.oetiker.ch/mrtg/>. Consultado em 27/11/2008
- [4] RRDTool. **RRDTool Official Web Site**, Disponível em <http://oss.oetiker.ch/rrdtool/>. Consultado em 27/11/2008
- [5] Nagios. **Nagios Official Web Site**, Disponível em <http://www.nagios.org>. Consultado em 27/11/2008
- [6] Cacti. **Cacti Official Web Site** Disponível em <http://cacti.net>. Consultado em 27/11/2008
- [7] Zabbix **Zabbix Official Web Site**, Disponível em <http://www.zabbix.org>. Consultado em 27/11/2008
- [8] Munin. **Munin Official Web Site**, Disponível em <http://munin.projects.linpro.no/>. Consultado em 27/11/2008
- [9] Zenoss. **Zenoss Official Web Site**, Disponível em <http://www.zenoss.com>. Consultado em 27/11/2008
- [10] Nav. **Nav Official Web Site**, Disponível em <http://metanav.uninett.no/>. Consultado em 27/11/2008
- [11] ANDRADE, H. A. **Nagios como solução de monitoramento de redes**, 2006. 65 p. Monografia – Departamento de Ciência da computação da Universidade Federal de Lavras, Lavras.
- [12] CAVA, R. A. **Bifocal Tree Uma Técnica Para Visualização de Estruturas Hierárquicas**. 2002. 97 p. Dissertação (Mestrado em Ciência da Computação) – Universidade Federal do Rio Grande do Sul, Porto Alegre.
- [13] Marcelo M. W., Pimenta, S, **Avaliação de Usabilidade de Sites Web**, 2001 LIHS - IRIT (*Institut de Recherche en Informatique de Toulouse*) 118, route de Narbonne 31062 - Toulouse Cedex 4 – França.

- [14] ANDREW S. **Redes de Computadores**. 5a.ed. tradução [3a. Ed. Americana].Ed. Campus Ltda. Rio de Janeiro, 1997.
- [15] CORREIA, L. H. A. et al. **Redes de Computadores**, Curso de Pós-Graduação “Lato Sensu” (Especialização) a Distância: Administração em Redes Linux – Lavras: UFLA/FAEPE, 2005.
- [16] TANENBAUM, Andrew S. **Redes de computadores** – 3. Ed. Rio de Janeiro: Campus, 2003.
- [17] CAMELO, Luiz David S. de P.**Modelo Para Gerenciamento Remoto De Falhas Em Arquiteturas De Ambientes Virtuais Em Redes**, 2006, 106p. Dissertação Apresentada ao Mestrado em Ciência da Computação Universidade Federal do Ceará.
- [18] LEINWAND, A. & FANG, K. **Network Management A Practical Perspective**, Addison- Wesley Publishing Company, USA, 1993.
- [19] STALLINGS, W. SNMP, SNMPv2, SNMPv3, and RMON 1 and 2. Prentice Hall. 1999.
- [20] BEVAN, N.**Usability is quality of use**, 1995. In: Anzai & Ogawa (eds) Proc. 6thInternational Conference on Human Computer Interaction, July. Elsevier.
- [21] NIELSEN, J. **Usability Engineering.**, 1993 Boston - USA: Academic Press, 362 p.
- [22] FREITAS, M. D. S. et al. **Introdução a Visualização de Informações**, Revista de Informática Teórica e Aplicada, vol. 3, nº 2, 2001-16p.
- [23] DO NASCIMENTO, H. A. D.; FERREIRA, C. B. R.. **Visualização de Informações – Uma Abordagem Prática**, 2005. XXV Congresso da Sociedade Brasileira de Computação, XXIV JAI. UNISINOS, S. Leopoldo – RS.
- [24] ALMEIDA, M.O. **Uma ferramenta para mineração visual de dados usando mapas em árvore e suas aplicações**, 2003, 59p, Dissertação (Mestrado Profissional em Redes de Computadores Computação) – Universidade de Salvador, Salvador.
- [25] FURNAS, George W. The FISHEYE View: A New Look at Structured Files. In: CARD, Stuart K.; MACKINLAY, Jock D.; SHNEIDERMAN, Ben. **Readings in Information Visualization: Using Vision to Think**. San Francisco, California: Morgan Kaufmann Publishers, 1999. p.312-330.
- [26] Lamping, J. and Rao, R.. **The hyperbolic browser: A focus + context technique for visualizing large hierarchies**, 1996. Journal of Visual Languages and Computing, 7(1):33–35.

- [27] EVANGELISTA. S. R. M, **Manual do HiperEditor e do HiperNavegador**. Campinas: Embrapa Informática Agropecuária, 2007, CDD-21° ed. 26p.
- [28] DA CRUZ. S. A. B, **Mecanismo para Edição de Árvore Hiperbólica através da Web**. Campinas: Embrapa Informática Agropecuária, 2007, CDD-21° ed. 26p.
- [29] BOUTHIER, C. **Bouthier family**. Disponível em: <http://www.bouthier.net> . Acesso em: Outubro de 2008.
- [30] THIOLENT, Michel. **Metodologia da pesquisa-ação**. São Paulo: Cortez: Autores Associados, 1986. 108p.