

REGINALDO JOSÉ ATISANO

**MONITORAMENTO DE REDES COM CACTI E
PHP NETWORK WEATHERMAP**

CURITIBA
2011

REGINALDO JOSÉ ATISANO

**MONITORAMENTO DE REDES COM CACTI E
PHP NETWORK WEATHERMAP**

Monografia apresentada ao Curso de Especialização em Informática com ênfase em Gerência de Redes de Computadores, Departamento de Informática, Setor de Ciências Exatas, Universidade Federal do Paraná, como requisito parcial para obtenção do título de Especialista em Informática.

Orientador: Professor Doutor Eduardo Todt

CURITIBA
2011

PARECER DE APROVAÇÃO


Monografia de Especialização em Informática
Ênfase em Gerência de Redes de Computadores
Programa de Pós-Graduação em Informática/UFPR

Declaramos que o aluno **REGINALDO JOSÉ ATISANO** entregou a versão final da sua Monografia de Especialização em Informática da Universidade Federal do Paraná, com Ênfase em Gerência de Redes de Computadores, intitulada **MONITORAMENTO DE REDES COM CACTI E PHP NETWORK WEATHERMAP**

Curitiba, 28 de outubro 2011



EDUARDO TODT
Professor Classe Adjunto I
Universidade Federal do Paraná
Setor de Ciências Exatas
Departamento de Informática
Caixa Postal 19081
CEP 81531-990 - Curitiba-PR



ANDREAS KIEFER
Professor Classe Assistente
Universidade Federal do Paraná
Setor de Ciências Exatas
Departamento de Informática
Caixa Postal 19081
CEP 81531-990 - Curitiba-PR

AGRADECIMENTOS

A Deus, acima de tudo, sem ele nada seria possível.

A minha esposa Lidia e meus filhos Lucas e Maria Clara pelo amor e compreensão em todos os momentos.

Aos meus pais, Maria e José Osvaldo, pelo amor incondicional e por me darem as melhores condições para o meu desenvolvimento, humano e intelectual.

A minhas irmãs Regina e Regiane, pelo amor, ajuda e colaboração em todos os momentos.

Ao professor Eduardo Todt, por sua orientação, auxílio, presteza e organização.

E a todos no Centro de Computação Eletrônica da Universidade Federal do Paraná, que de várias formas me auxiliaram neste trabalho.

RESUMO

Este trabalho apresenta de forma simples e direta o *PHP Network Weathermap*, um dos principais *plugins* do *Cacti*, que por sua vez é um importante software de monitoramento de redes. Esta ferramenta permite transformar os dados coletados pelo *Cacti* em mapas, desta maneira, centralizando todas as informações da rede gerenciada em uma única página. Isto possibilita um amplo controle da rede, habilitando o administrador de rede a detectar com facilidade falhas, tráfegos maliciosos, pontos críticos, e principalmente, lhe possibilita ter uma visão geral da saúde de toda a rede. Seu desempenho foi analisado em uma rede de grande porte, a rede da Universidade Federal do Paraná, onde o *PHP Network Weathermap* pode demonstrar suas principais características e vantagens, trazendo à tona recursos essenciais ao monitoramento de redes, mas que ainda não haviam sido apresentados adequadamente. Assim, o *PHP Network Weathermap* se credencia como uma importante ferramenta para os profissionais da área de TI. Por meio dos resultados ficou clara a contribuição que este projeto *open source* presta à gerência de redes. Este trabalho também busca divulgar aos profissionais da área o quão é necessário o monitoramento de rede para todas as empresas ou corporações, independente do seu porte ou ramo de atividade.

Palavras-chave: *PHP Network Weathermap*. *Cacti*. Monitoramento de redes. Gerenciamento de redes.

ABSTRACT

This paper presents a simple and direct Weathermap PHP Network, a leading Cacti plugins, which in turn is important network monitoring software. This tool allows you to transform the data collected by Cacti on maps, thus centralizing all information on the managed network in a single page. This allows full control of the network, enabling the network administrator to easily detect failures, malicious traffic, points critics, and most importantly, it affords an overview of the health of the entire network. Its performance was analyzed in a large network, the network of the Federal University of Parana, where PHP Network Weathermap could demonstrate its main features and benefits, bringing out the essential features of monitoring networks, but that had not been presented adequately. Thus it is accredited as an important tool for professionals in the IT field. Through the results became clear the contribution that this open source project provides the management of networks. This work also seeks to disseminate to the professionals the relevance of monitoring the network for all companies or corporations, regardless of their size or industry.

Key words: PHP Network Weathermap. Cacti. Monitoring networks. Network management.

LISTA DE ILUSTRAÇÕES

Figura 1 – Modelo de Gerência de Redes com uso do protocolo	15
Figura 2 – Exemplo de mapa do PHP Network Weathermap.....	17
Figura 3 – Exemplo de visualização de gráficos a partir do link	19
Figura 4 – Exemplo cedido por Simon Schwendemann.....	19
Figura 5 – Mapa rede da University of Texas at San Antonio	20
Figura 6 – Exemplo cedido por Mike Nipp	20
Figura 7 - Tela principal do Editor Gráfico	27
Figura 8 – Exemplo de tela de propriedade para editar Nó	29
Figura 9 – Exemplo de tela de propriedade para editar Link	29
Figura 10 – Exemplos de texto de arquivo de configuração.....	31
Figura 11 – Guia visual do PHP Network Weathermap	34
Figura 12 – Exemplo de arquivo de configuração de mapa	46
Figura 13 – Exemplo desenvolvido pelo autor do manual, Howard Jones	46

LISTA DE TABELAS

TABELA 1 – Comparativo antes e depois do monitoramento da rede UFPR.....	35
TABELA 2 – Dados analisados no Cacti	37

SUMÁRIO

1 INTRODUÇÃO	10
1.1 CACTI	11
1.2 MOTIVAÇÃO	12
1.3 OBJETIVOS E METODOLOGIA	13
2 CACTI	14
2.1 CARACTERÍSTICAS DO CACTI.....	14
2.2 PHP NETWORK WEATHERMAP	17
2.2.1 Histórico do PHP Network Weathermap.....	21
2.2.2 Principais características do PHP Network Weathermap.....	21
2.2.3 Evolução das versões do PHP Network Weathermap.....	22
3 PHP NETWORK WEATHERMAP E O MONITORAMENTO DE REDES	26
3.1 PRINCIPAIS VANTAGENS DO PHP NETWORK WEATHERMAP	26
3.1.1 Recursos disponíveis por meio do editor gráfico.....	26
3.1.2 Recursos disponíveis por meio do editor de texto.....	30
3.1.3 Principais configurações	32
3.1.4 Comparativo de demandas do ambiente monitorado.....	35
3.2 CACTIEZ.....	36
4 ANÁLISE DOS RESULTADOS.....	37
4.1 DADOS OBTIDOS	37
4.2 PONTOS POSITIVOS.....	38
4.3 PONTOS NEGATIVOS.....	39
4.4 CONSIDERAÇÕES FINAIS	39
5 CONCLUSÃO	40
REFERÊNCIAS BIBLIOGRÁFICAS	41
ANEXO	43

1 INTRODUÇÃO

A década de 60 marcou o início efetivo do desenvolvimento das redes de computadores, pois somente então é que ocorreram condições propícias para iniciar o seu real desenvolvimento. Existiram experiências anteriores, mas tinham aplicações restritas em atividades militares (KUROSE, 2006). A partir do final dos anos 60, com a evolução dos computadores, motivada principalmente pela redução de seus componentes internos, aumento do seu poder de processamento e redução dos custos de produção, ocorreu a sua difusão em todas as áreas da sociedade. Com a crescente utilização dos computadores em pontos geograficamente distantes, naturalmente surgiu a necessidade de conectá-los, pois a quantidade de informações crescia e precisava ser compartilhada e acessada remotamente.

Em resposta a esta necessidade surgiram as redes de computadores. Elas não começaram de forma única e coordenada, houve vários projetos, que paralelamente foram desenvolvidos, mas que ao longo de anos foram convergindo para uma solução mais homogênea. Podemos citar como principais projetos a *ARPANet*, desenvolvida a partir de 1967 pela ARPA (*Advanced Research and Projects Agency*), rede, a princípio para uso militar do Departamento de Defesa dos EUA; a *ALOHAnet* (Universidade do Hawaii), primeira rede via rádio, criada em 1970; e, a *Minitel*, criada em 1982 e tendo como principal função levar o serviço de vídeo texto aos lares franceses (TANENBAUM, 2003).

A partir do crescimento das redes de computadores surgiu a necessidade de colher informações dos equipamentos e de todos os componentes físicos que formavam esta rede, ou seja, gerenciar os ativos de rede. No princípio e até a metade dos anos 80, as soluções que predominaram no mercado eram as soluções proprietárias. Cada empresa detentora da tecnologia da rede, também promovia soluções de gerenciamento, mas cada qual com seu protocolo específico, o que as tornavam caras e ineficientes. Como o mercado de computadores e, conseqüentemente, de equipamentos de rede crescia exponencialmente e não havia um padrão entre os diversos fabricantes, foi criado um comitê técnico em comunicação de Dados, chamado IFIP (*International Federation for Information Processing*) (STALLINGS, 1999), o qual começou a trabalhar em busca de um consenso entre os diversos fabricantes de *hardware*.

Após um longo período de negociações, chegou-se a uma definição. Em 1988 o protocolo SNMP (*Simple Network Management Protocol*), que já era uma evolução do SGMP (*Simple Gateway Monitoring Protocol*), criado em 1987, foi adotado temporariamente como padrão. Para longo prazo, seria adotado o protocolo CMOT (CMIP over TCP/IP), fato que não se concretizou, pois a simplicidade e praticidade do protocolo SNMP foi rapidamente aceita por todos, desta forma sendo ele definitivamente o padrão. Sofreu evoluções visando maior segurança, como o SNMPv2, SNMPv2c e SNMPv3 com criptografia, a versão atual (STALLINGS, 2005).

A partir do estabelecimento de um padrão de protocolo de gerenciamento, a indústria de *hardware* e a de *software* de rede voltada ao gerenciamento começou a se desenvolver rapidamente e, principalmente, a oferecer opções não proprietárias, pois agora os novos equipamentos suportavam a mesma linguagem de comunicação. Com isto, estas novas plataformas de gerenciamento poderiam ter maior abrangência, desempenho e proporcionar melhor controle por parte do administrador de rede.

1.1 CACTI

Por que monitorar? Atualmente parece óbvio, mas no início, no surgimento das redes, não foi assim. Provavelmente pela novidade da tecnologia ou por desconhecimento do quão abrangente pode ser sua área de atuação e, conseqüentemente, quais atividades podem ser afetadas, principalmente quando da ocorrência de eventos inesperados, como por exemplo, uma falta de energia. Um evento deste tipo pode ocasionar o atraso ou, até mesmo, a paralisação de inúmeras atividades primordiais em uma empresa ou corporação, visto que hoje em todas as áreas profissionais há a necessidade de aplicações via *Web* (STALLINGS, 2005).

Em resposta a estes questionamentos surgiram as ferramentas de monitoramento, pois para quem gerencia uma rede de computadores, o principal objetivo é mantê-la 100% do tempo em disponibilidade. Claro que esta é uma meta ideal, hoje se trabalha com patamares reais entre 99,0% e 99,9% do tempo com disponibilidade dos recursos. Independente do ramo de atividade da empresa, esta disponibilidade poderá se traduzir em lucro ou prejuízo, sendo este um dos principais motivos que levam as corporações a monitorar suas redes (COSTA, 2008).

Após a definição do protocolo SNMP como padrão para o gerenciamento de redes pelos principais fabricantes do mundo, no final da década de 80, o desenvolvimento de novas ferramentas de gerenciamento ganhou força e atingiu uma abrangência muito grande, pois o protocolo SNMP permitiu que *routers*, *switches* e demais ativos de rede fossem monitorados, independentemente de quem o fabricou (KUROSE, 2006). Este novo paradigma permitiu que as empresas se libertassem do monopólio do gerenciamento proprietário, que era o modelo vigente até aquele momento. Neste mesmo período outra grande mudança estava em andamento, o desenvolvimento de aplicações a partir de código aberto, *open source*. Todos estes fatores propiciaram um ambiente favorável para a criação de novas ferramentas voltadas ao gerenciamento de redes. É neste clima de aplicações livres e sem limitações, tanto em relação a seu desenvolvimento como de sua abrangência de utilização, que surge o *Cacti*, no final da década de 90.

Sua primeira versão como produto surgiu em setembro de 2001. Desde então, vem se desenvolvendo rapidamente e evoluindo juntamente com as redes as quais se propõe a monitorar. Muito do seu sucesso se deve principalmente ao seu desenvolvimento *open source*, o que permite que vários desenvolvedores adicionem sua contribuição. Seus principais colaboradores até o momento são Ian Berry, Larry Adams, Tony Roman, Jimmy Conner, Reinhard Scheck e Jp Pasmak (SOURCEFORGE, 2011).

O seu ponto forte é a possibilidade de gerar gráficos sobre todos os dados coletados, como por exemplo: gráficos de largura de banda, temperatura, espaço em disco, utilização de CPU, permitindo inclusive personalização e histórico. Além disso, oferece ao usuário uma interface prática, de fácil utilização, com desenvolvimento constante e configuração intuitiva. Desta forma, tornou-se um dos mais populares *softwares* de monitoramento de redes entre os administradores de rede.

1.2 MOTIVAÇÃO

Apesar de ser consideravelmente difundido entre os administradores de rede, o *Cacti* é ainda muito utilizado em sua configuração básica, ou seja, sem os *plugins*. Isto o torna uma ferramenta sem um diferencial efetivo. A utilização de seus principais *plugins*, e principalmente o *PHP Network Weathermap*, o colocam como

um conjunto de ferramentas de extrema utilidade no monitoramento das redes de computadores.

Por meio das pesquisas bibliográficas nos portais ACM, IEEE Xplore, livros, *sites* de fabricantes e comunidades de desenvolvedores, concluímos que existe uma lacuna referente à análise e divulgação do *PHP Network Weathermap*. Portanto, este trabalho visa trazer à luz as qualidades e características deste importante *plugin* para a área do gerenciamento de redes, contribuindo, desta forma, para suprir esta ausência de informações a respeito deste importante *software*.

Os estudos e avaliações aqui descritos foram todos desenvolvidos sobre a versão 0.95b do *PHP Network Weathermap*. A rede analisada foi a da Universidade Federal do Paraná, uma rede de grande porte, com mais de 4500 pontos.

1.3 OBJETIVOS E METODOLOGIA

Este trabalho tem como principal objetivo apresentar uma das principais ferramentas disponíveis para o *Cacti*, o *plugin PHP Network Weathermap*, que é um recurso extremamente importante para visualização e controle de toda a rede, mas que ainda não é adequadamente conhecido e explorado pela comunidade de TI.

A princípio são descritas as funcionalidades do *Cacti*, bem como as características do *plugin PHP Network Weathermap* detalhadamente. Em seguida são demonstrados, através de experimentos e resultados, seus principais recursos, características, pontos positivos e negativos, sempre levando em consideração as reais dificuldades de utilização em um ambiente de redes, principalmente, em uma estrutura de grande porte.

2 CACTI

Este capítulo apresenta uma análise das origens desta importante ferramenta, seu funcionamento, suas vantagens e também destacando seus principais *plugins*. Depois é focalizado com especial atenção o *plugin PHP Network Weathermap*, salientando suas características e descrevendo seu desenvolvimento e evolução através de suas principais versões.

2.1 CARACTERÍSTICAS DO CACTI

O *Cacti* é um *Frontend* para a ferramenta *RRDTool* (*Round Robin Database Tool*) que coleta e organiza os dados (OETIKER, 2011). Ela permite a captura de dados de diversos dispositivos de rede, sem gerar um volume excessivo de dados armazenados, pois utiliza uma técnica que permite o armazenamento dos dados de forma otimizada. Isso é possível porque a *RRDTool* adota métricas baseadas na média dos dados e não pelo seu total. Por exemplo, para os dados mais antigos ela somente armazena a média dos valores, assim, consegue ter um histórico considerável de dados, sem que isso acarrete um número exagerado de informação, fato esse que pode vir a demandar muito espaço de armazenagem como também tornar lenta a ferramenta que irá utilizar estes dados (OETIKER, 2011). Este é o principal motivo que permite ao *Cacti* monitorar muitos dispositivos de rede, com pouco impacto em seu desempenho.

Para armazenar estes dados o *Cacti* utiliza o banco de dados *MySQL*, uma ferramenta atual que já se mostrou confiável para cumprir as necessidades do sistema.

Quanto à interface *WEB*, utiliza o padrão *PHP*, o que permite um grau elevado de compatibilidade com os principais navegadores do mercado, como também um desempenho adequada para os padrões do sistema.

O grande diferencial do *Cacti* é a possibilidade de gerar diversos tipos de gráficos com base nos dados coletados e, principalmente, de forma customizada. Como exemplo, é possível gerar gráficos de largura de banda, temperatura e uso de CPU, possibilitando ao usuário uma visão geral dos dispositivos e serviços gerenciados (CACTI, 2009).

De forma simplificada, o funcionamento do *Cacti* se realiza por meio da coleta de dados dos dispositivos cadastrados. Esta operação, denominada *polling*, se utiliza principalmente do protocolo SNMP, para receber dos equipamentos e serviços as principais informações que serão armazenadas no banco de dados *MySQL*. Este armazenamento ocorre com a ferramenta *RRDTool*. Com o uso desta técnica há um aumento mínimo na base de dados, levando-se em consideração a grande ocorrência de *pollings*, que dependendo da quantidade de dispositivos monitorados, pode ocorrer com frequência inferior a um minuto. Isto permite a coleta de dados de uma elevada quantidade de dispositivos de rede. Em seguida, a interface *Web* se utiliza desta base de dados para gerar gráficos, que por sua vez são configurados de acordo com as necessidades de cada usuário, oferecendo possibilidades diversas no monitoramento das redes. A Figura 1, logo a seguir, exemplifica um caso de gerência de rede, onde o gerente (Servidor de gerência) se comunica com os diversos tipos de agentes (*switches*, roteadores, hosts) coletando as informações:

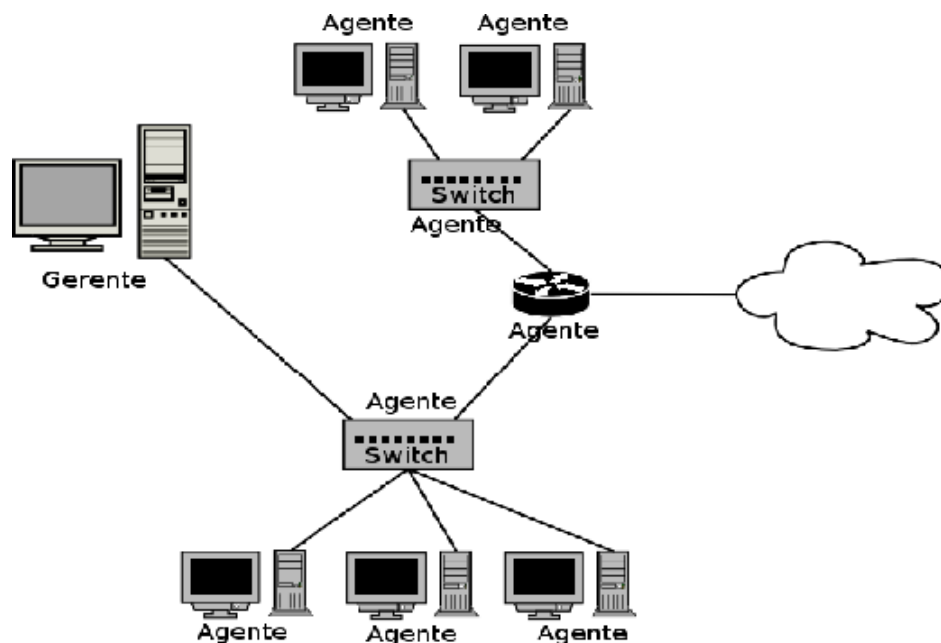


Figura 1 – Modelo de Gerência de Redes com uso do protocolo SNMP. (MATOS, 2009)

O *Cacti* possui uma arquitetura aberta, por isso permite a adição de *plugins* que possibilitam incrementar ainda mais seu uso e, conseqüentemente, seus resultados. Segue uma breve descrição dos principais *plugins* disponíveis atualmente (CONNER, 2011-A):

- *Thold* → Permite a criação de limites para os valores dos dados dos gráficos, depois de estabelecidos estes limites, caso haja o rompimento dos mesmos, é

gerado um alerta através de *e-mail*, avisando sobre a possível falha apresentada. *Plugin* extremamente útil, pois permite a tomada de ação rápida por parte do administrador da rede. Por exemplo, digamos que o limite máximo para a temperatura de determinado equipamento é de 50 graus *Celsius*, estabelece-se então este limite. Caso haja um superaquecimento e este limite seja ultrapassado, o *Thold* irá gerar um alerta via *e-mail* para o administrador;

- *Monitor* → Possibilita ver e ouvir alertas através de uma interface *Web*, pela qual são mostrados todos os equipamentos monitorados. Caso algum saia de operação, haverá uma sinalização através de cores padronizadas em alertas sonoros;
- *Discovery* → Habilitando este *plugin* o usuário do *Cacti* pode obter uma relação com todos os dispositivos de rede que ainda não foram cadastrados, mas que pertencem a sua rede e adicioná-los ao monitoramento;
- *Realtime* → Este *plugin* otimiza a visualização dos gráficos, pois permite exibí-los em tempo real, podendo atualizar os dados com intervalos de até 5 segundos. Muito útil na manutenção de equipamentos e na localização de pontos de congestionamento na rede;
- *Nmidsmokeping* → Habilita a ferramenta *Smokeping* dentro do *Cacti*. Esta ferramenta apresenta através de gráficos a latência na resposta de equipamentos e, principalmente de *links*. Muito utilizada para detectar lentidão nos *links*;
- *Weathermap* → Ativa o *PHP Network Weathermap*, uma das mais úteis ferramentas, pois exhibe através de mapas todas as informações da rede monitorada, apresentando de forma sintética a saúde dos dispositivos e *links*.

Existem muitos outros *plugins* disponíveis para o *Cacti*, mas estes são os mais utilizados. A seguir, é abordado especificamente um deles, o *PHP Network Weathermap*. Esta importante ferramenta de apoio ao gerenciamento de rede, apesar de já estar disponível há vários anos, ainda permanece com muitas de suas características pouco conhecidas. Seu diferencial é a capacidade de reunir todas as informações referentes à rede em um único local, um único mapa, desta forma permitindo ao administrador de rede tomar decisões rápidas, tanto na detecção como na solução de falhas ou problemas.

2.2 PHP NETWORK WEATHERMAP

O *PHP Network Weathermap* é um *software* desenvolvido para permitir a visualização de *links* de dados, roteadores, *switches*, gráficos e demais itens que compõem uma rede de computadores. Isto é possível, com sua instalação como *plugin* de uma ferramenta de monitoramento, ou seja, como um recurso adicional. Exemplo de ferramenta de monitoramento é o *Cacti*, mas poderia ser qualquer outra ferramenta, desde que se utilize do *RRDTool*, pois este recurso, como já descrevemos anteriormente, permite a coleta de dados e seu armazenamento de forma otimizada, além de ser pré-requisito para este uso compartilhado. No caso do *Cacti*, o *PHP Network Weathermap* se utiliza desta base de dados, para, por meio de mapas, exibir de forma sumarizada a situação de toda rede, e o que é melhor, em tempo real. Para o administrador de rede é possível observar, por exemplo, quais *links* estão ativos, qual o seu nível de utilização, quais equipamentos ou redes estão em operação. Isto possibilita uma visão global de toda a rede, oferecendo a possibilidade de identificar e agir de forma mais rápida e eficaz em relação aos possíveis problemas detectados. Segue, na Figura 2, exemplo de Mapa do *PHP Network Weathermap* que sintetiza bem esta ideia.

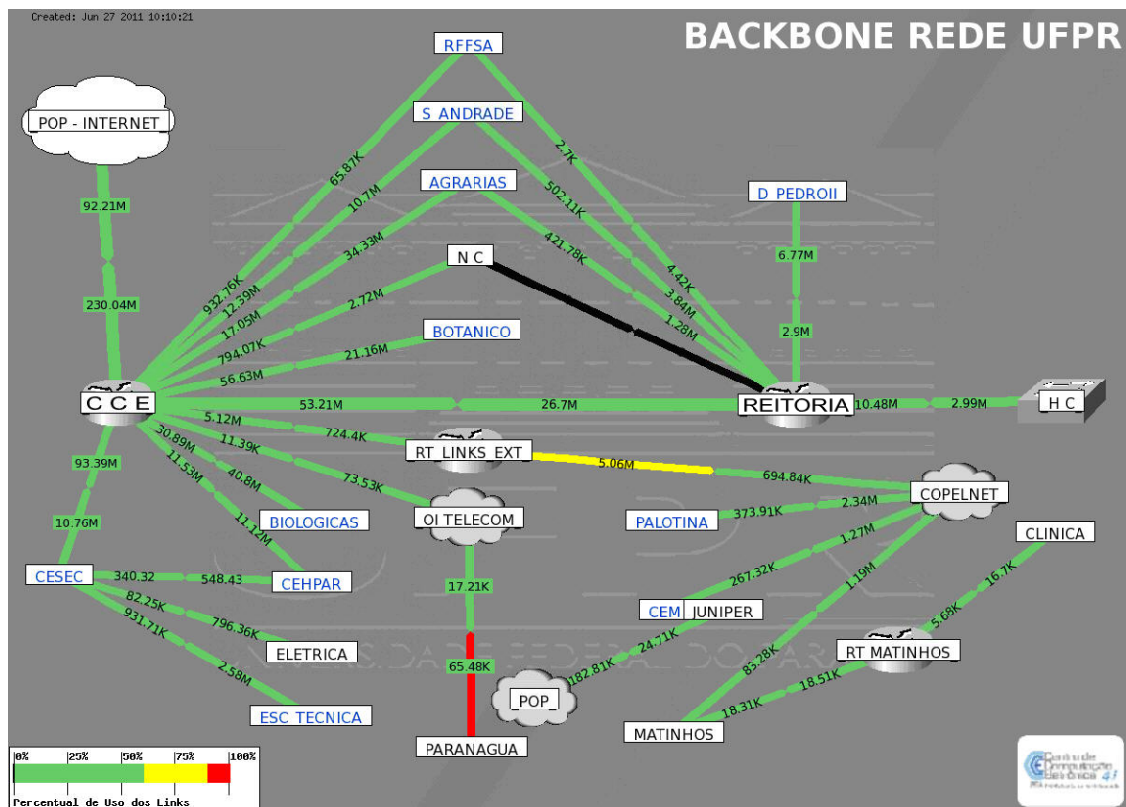


Figura 2 – Exemplo de mapa do PHP Network Weathermap. (CCE, 2011)

Na figura 2 podemos observar várias informações que são fundamentais para o administrador da rede ter o controle dos equipamentos, *links* e do tráfego da rede:

- Cada retângulo branco com um nome simboliza um nó (nós são normalmente equipamentos como roteadores ou *switches*) este nó pode representar um roteador, *switch* ou mesmo um servidor;
- Os desenhos em formato de nuvem representam o acesso a *Internet*, normalmente é conectado a uma operadora de telecomunicação;
- Os *links* são as linhas que, em sua maioria, estão na cor verde. Eles representam as conexões entre os nós;
- A forma de seta no meio dos *links* indica o sentido do tráfego de dados. Quando aponta no sentido oposto do nó ao qual está conectado, indica tráfego de saída (*outbound*), quando aponta no mesmo sentido, indica tráfego de entrada (*inbound*);
- Os números no interior do *link* indicam a quantidade de dados por segundo que estão trafegando naquele momento no *link*. O número que está na parte do *link* com a seta apontando no sentido oposto ao do nó, indica qual é a largura da banda de saída naquele momento e, o número que está na posição no sentido do nó, é a largura da banda de entrada;
- No canto inferior esquerdo do mapa existe uma legenda com uma escala de cores. Ela permite a leitura do tráfego dos *links*, variando as cores de acordo com a utilização dos *links*. Neste caso a escala está configurada da seguinte forma. De 0,1 % a 60% de tráfego os *links* permanecem na cor verde (normal), de 60,1% a 90% mudam para amarelo (alerta) e de 90,1% a 100% mudam para cor vermelha (crítico). Além disso, quando não existe tráfego no *link* ele assume a cor preta. Desta forma, sinalizam em que níveis de utilização estão os *links*;
- Os nós que estão com os nomes na cor azul permitem o acesso a outros mapas, que detalham as conexões internas dos nós acessados;
- A taxa de atualização das informações no mapa pode ser configurada, podendo variar de cinco segundos a cinco minutos;
- Outro recurso útil nos mapas do *PHP Network Weathermap* é acessar os gráficos referentes a cada *link*, simplesmente clicando sobre o *link* desejado, obtendo assim, os gráficos completos ou somente posicionando

o cursor sobre o *link*, quando é exibido um gráfico em uma versão reduzida. Como é exemplificado na figura 3, abaixo.

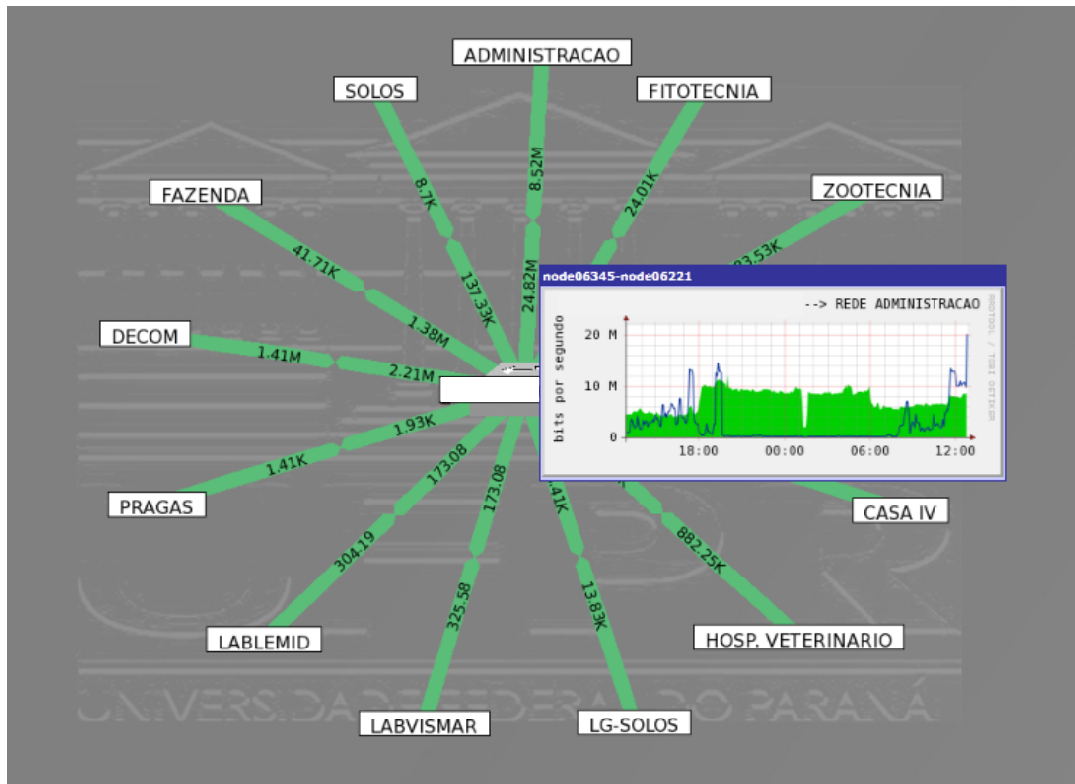


Figura 3 – Exemplo de visualização de gráficos a partir do link. (CCE, 2011)

A seguir, nas figuras 4, 5 e 6 são apresentados mais exemplos de mapas do *PHP Network Weathermap*, demonstrando as várias possibilidades de configurá-lo:

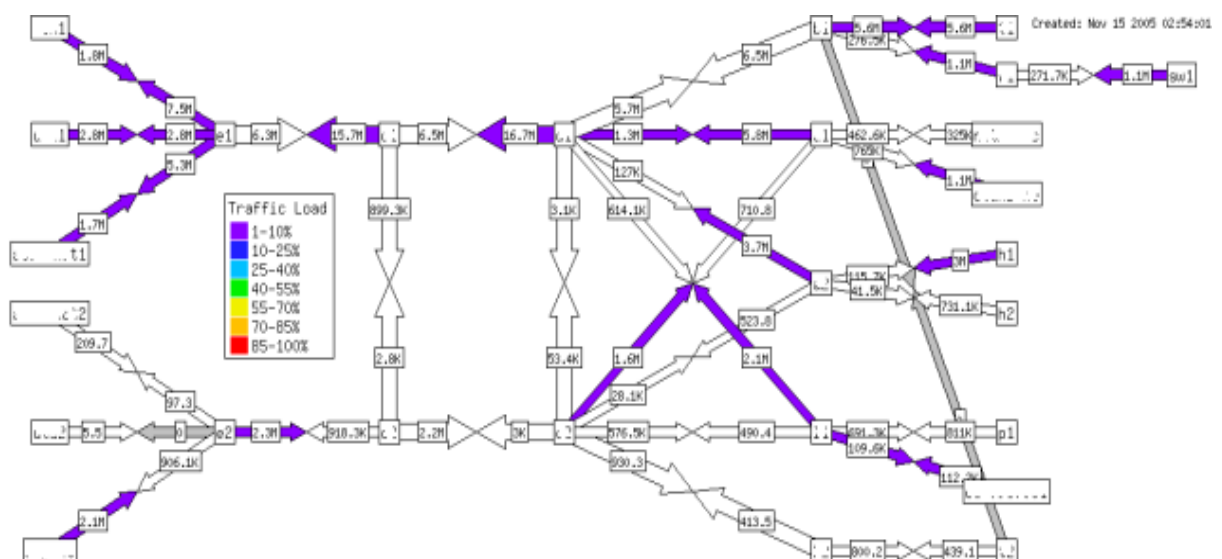


Figura 4 – Exemplo cedido por Simon Schwendemann (HOWARD, 2010-B)

2.2.1 Histórico do PHP Network Weathermap

O *PHP Network Weathermap* surgiu em abril de 2005, em sua versão 0.1. Seu desenvolvimento, desde o início, está a cargo do britânico Howard Jones, porém, ele mesmo observa que a ideia não é originalmente sua, utilizou como ponto de partida vários outros projetos com propostas semelhantes (HOWARD, 2010-A). O principal deles foi o *Network Weathermap* criado pelo grego Panagiotis Christias. O princípio é o mesmo, ou seja, centralizar em um ou vários mapas todos os dispositivos e principais informações referentes à rede. A ferramenta criada por Christias foi desenvolvida na linguagem de programação PERL, enquanto nosso objeto de estudo utiliza como plataforma o *PHP*. Além disso, a criação de Christias não prosseguiu, em meados de 2006 cessaram os trabalhos de desenvolvimento (CHRISTIAS, 2006).

Howard fez uma boa aposta, pois a linguagem *PHP* se desenvolveu e se popularizou. Tanto o primeiro quanto o segundo projeto são iniciativas *open source* e, portanto, permitem modificações e melhorias constantes (HOWARD, 2010-A).

2.2.2 Principais características do PHP Network Weathermap

- Projeto estável. É um projeto que se mantém há vários anos e sempre oferecendo atualizações e melhorias, suprimindo assim, as demandas apresentadas;
- Integração. O *PHP Network Weathermap* possui total compatibilidade com o *Cacti*, oferecendo aproveitamento de todos os dados coletados e sumarizando em mapas de fácil leitura e configuração;
- Flexibilidade. Permite diversas configurações para mapas, nós, *links* e demais objetos inseridos. Isto coloca à disposição do usuário diversas possibilidades para exibir e gerenciar sua rede, seja qual for a topologia de rede;
- Editor Gráfico. O *Weathermap Editor (Web)* facilita a configuração tanto de novos mapas quanto a atualização dos já existentes, podendo ser utilizado em conjunto com o editor de texto. Ele permite alterações como alterar a aparência de nós e *links*, curvar *links*, adicionar dados de várias fontes;

- Compatibilidade. O *PHP Network Weathermap* já foi testado para os seguintes sistemas operacionais: *Linux*, *FreeBSD*, *Mac OS X* e *Windows*. Além disso, pode ser utilizado em conjunto com outros *softwares* como o *Cacti*, como é o caso do *MRTG*, ou qualquer outro que utilize a ferramenta *RRDTool*;
- Baixo Custo. Além de possuir licença de *Software Livre* (Licença Pública Geral GNU, Versão 2), a curva de aprendizagem é pequena, se comparada com ferramentas similares no mercado atual (HOWARD, 2010-A).

2.2.3 Evolução das versões do PHP Network Weathermap

Neste ponto, são descritas as principais características do *PHP Network Weathermap* de forma mais detalhada, traçando sua evolução através de uma linha do tempo, que contempla as suas principais versões.

Devemos ressaltar que em suas primeiras versões, o *PHP Network Weathermap* sofreu várias atualizações, visando essencialmente sanar *bugs*, o que se tratando de um novo *software* é aceitável. No início surgiram problemas diversos, desde incompatibilidade com determinadas aplicações até falhas relativas à sua própria execução. Portanto, das versões 0.1 (Abril/2005) a 0.82 (Maio/2006) as ações foram neste sentido. Existem alguns aspectos das versões iniciais a serem ressaltados, como por exemplo, na versão 0.4 (Maio/2005) a elaboração do primeiro manual para o usuário. Nesta mesma versão foi alterada a forma de exibição da leitura de banda dos *links* para *bits*, permitindo assim um alinhamento com os padrões mundialmente utilizados. Na versão 0.7 (Dezembro/2005) houve uma profunda atualização do manual do usuário. A partir da versão 0.9 (Abril/2007) observaram-se grandes avanços para o usuário.

Versão 0.9 (Abril/2007) → Nesta versão os avanços foram:

- Possibilidade de adicionar comentários aos *links*;
- Mobilidade dos rótulos da largura de banda nos *links*, permitindo uma melhor organização e clareza das informações;
- Suporte a imagens no formato JPEG e GIF;

- É nesta versão que surge o recurso que permite mudança da escala de cores de acordo com o percentual de utilização dos *links*, alertando de forma mais objetiva quanto a uma possível saturação dos *links*.

Versão 0.91 (Maio/2007) → Esta versão foi gerada principalmente para corrigir *bugs* da versão 0.9, mas houve alguns progressos:

- Criado o *Quiet Mode Log*. Esta funcionalidade permite uma redução considerável dos *logs*, pois somente são registrados os *logs* no padrão *LOW* (somente registra *logs* de estatística e erros, mantendo assim uma baixa quantidade de registros);
- A subtração agregada permite a separação de tráfegos específicos na leitura da largura de banda. Por exemplo, em uma determinada interface é possível separar o tráfego normal do tráfego de uma VPN.

Versão 0.92 (Junho/2007) → O forte desta versão são novas funcionalidades do editor *Weathermap*:

- Funções como a *BWSTYLE* que possibilita o alinhamento da caixa *BWLABEL* com os *links*;
- A função *COMMENTPOS*, permite a mobilidade dos comentários;
- Também é criada nesta versão a possibilidade de adicionar um *link* a um nó, isso faz com que a partir deste nó se atinjam outros nós ou mesmo outras páginas de mapa. Além disso, houve correções referentes à versão anterior.

Versão 0.93 (Outubro/2007) → Nesta, as principais novidades ficaram por conta de várias pequenas melhorias na edição dos objetos inseridos nos mapas:

- Permite mostrar as coordenadas de localização dos nós, tanto em sua criação como em seu ajuste no mapa, facilitando em muito sua manipulação;

Versão 0.94 (Novembro/2007) → Nesta versão, novas facilidades na edição:

- Criação da ferramenta *LINKSTYLE* que permite a formatação dos *links* em forma de seta;
- Também a opção *RRD DS* permitiu um melhor desempenho dos *pollings*, principalmente para redes maiores, pois passou a utilizar o *plugin Boost*, criado especificamente para otimizar grandes quantidades de *polling*;

- Houve a integração das bases de dados com os novos *plugins Thold* e *Monitor*, o que permitiu a geração de mensagens de alerta quando quebrados os limites das variáveis previamente programadas para estes *plugins*.

Versão 0.95b (Março/2008) → A principal novidade da versão 0.95b foi uma otimização em suas linhas de comando, o que o tornou mais rápido e confiável:

- Foi criada a funcionalidade LABELANGLE que permitiu a colocação em ângulos dos títulos dos nós.

Versão 0.96 (Abril/2009) → Foram realizadas várias modificações na base de dados RRD, permitindo assim um melhor desempenho no *polling*:

- Funcionalidades para a edição dos mapas foram adicionadas, como por exemplo: ESCALE (define um percentual dentro da escala de cores), TARGET (especifica onde procurar as informações) e KEYSTYLE (especifica em que posição ficará a legenda, na horizontal ou na vertical).

Versão 0.97 (Janeiro/2010) → Trouxe como principal inovação a criação de um seletor que permitiu a navegação entre mapas, o qual possibilita a exibição destas imagens em tamanho real:

- Outra novidade foi o novo *plugin Datasource*, que remodelado, permitiu a geração de dados estatísticos indisponíveis anteriormente. Exemplo disso são as estatísticas diárias, semanais, mensais e até anuais.

Versão 0.97a (Março/2010) → Esta é a atual versão do *PHP Network Weathermap*:

- Trouxe como inovação a ferramenta LINK WIDTH, que agora permite casas decimais.
- Outro recurso, o CACTI-INTEGRATE.PHP possibilitou gerar DSStats (estatística) para a ferramenta TARGET;
- Mas a principal novidade foi a atualização para que a base de dados aceitasse dados de largura de banda em formato acima de 1Gbps, fato esse que já estava causando uma defasagem em relação aos padrões atuais (HOWARD, 2010-A).

Algo que vale a pena destacar em todo o processo de desenvolvimento do *PHP Network Weathermap*, é que Howard sempre fez questão de informar e nominar cada um dos desenvolvedores que colaboraram de uma forma ou outra, permitindo assim que muitos avanços fossem a ele adicionados. Foram estes avanços que permitiram a esta ferramenta atingir os seus objetivos e, conseqüentemente, satisfazer as demandas apresentadas por seus usuários.

No próximo capítulo é abordado o *plugin PHP Network Weathermap*, onde são demonstrados em detalhes, particularidades de seu funcionamento.

3 PHP NETWORK WEATHERMAP E O MONITORAMENTO DE REDES

Neste capítulo o *PHP Network Weathermap plugin* para o *Cacti*, é abordado em detalhes, sendo consideradas suas vantagens, principais recursos, configurações e pontos que ainda permitem melhorias.

3.1 PRINCIPAIS VANTAGENS DO PHP NETWORK WEATHERMAP

O *PHP Network Weathermap* é uma solução atraente para o monitoramento de redes de computadores, pois consegue sintetizar em um ou mais mapas a saúde de toda a rede. Destacamos abaixo, algumas destas vantagens observadas em sua utilização:

- Possibilita se antecipar ao usuário na localização e solução de falhas;
- Alertas visuais sobre falhas, através da escala de cores aplicada aos *links*;
- Identifica *links* fora de operação;
- Mede a largura de banda dos *links*;
- Detecta tráfego suspeito/malicioso, por meio da análise da largura de banda;
- Oferece uma visão geral quanto à localização e abrangência de falhas ou problemas;
- É uma referência para dimensionar e ampliar *links* e redes com largura de banda próximos da saturação;
- Permite configuração personalizada, conforme as necessidades de cada rede.

A seguir, de forma mais detalhada, passamos a demonstrar as configurações básicas do *PHP Network Weathermap*.

3.1.1 Recursos disponíveis por meio do editor gráfico

Para editar os arquivos de configurações do *PHP Network Weathermap* é possível utilizar o editor gráfico e o editor de texto.

O editor gráfico é uma versão *Web* do editor de texto e atua de forma complementar a este. Já para editar texto, utilizam-se normalmente editores como o VI ou VIM. Portanto, para usufruir dos recursos disponíveis nesta ferramenta, é necessário ter alguns conhecimentos básicos de *Linux*. Os arquivos que necessitam ser editados normalmente se localizam no diretório `/WEATHERMAP/CONFIGS` (o caminho completo na Distribuição *CentOS* 4.7 é: `/var/www/html/plugins/weathermap/configs/`). Cada mapa criado no editor gráfico gera um novo arquivo de texto cujo nome assume o formato "NOMEMAPA.CONF". Desta maneira, para os recursos que não podem ser adicionados/alterados pelo editor gráfico, há a necessidade de editá-los através dos arquivos de configurações. Segue, na Figura 7, a página inicial do editor gráfico do *PHP Network Weathermap*:

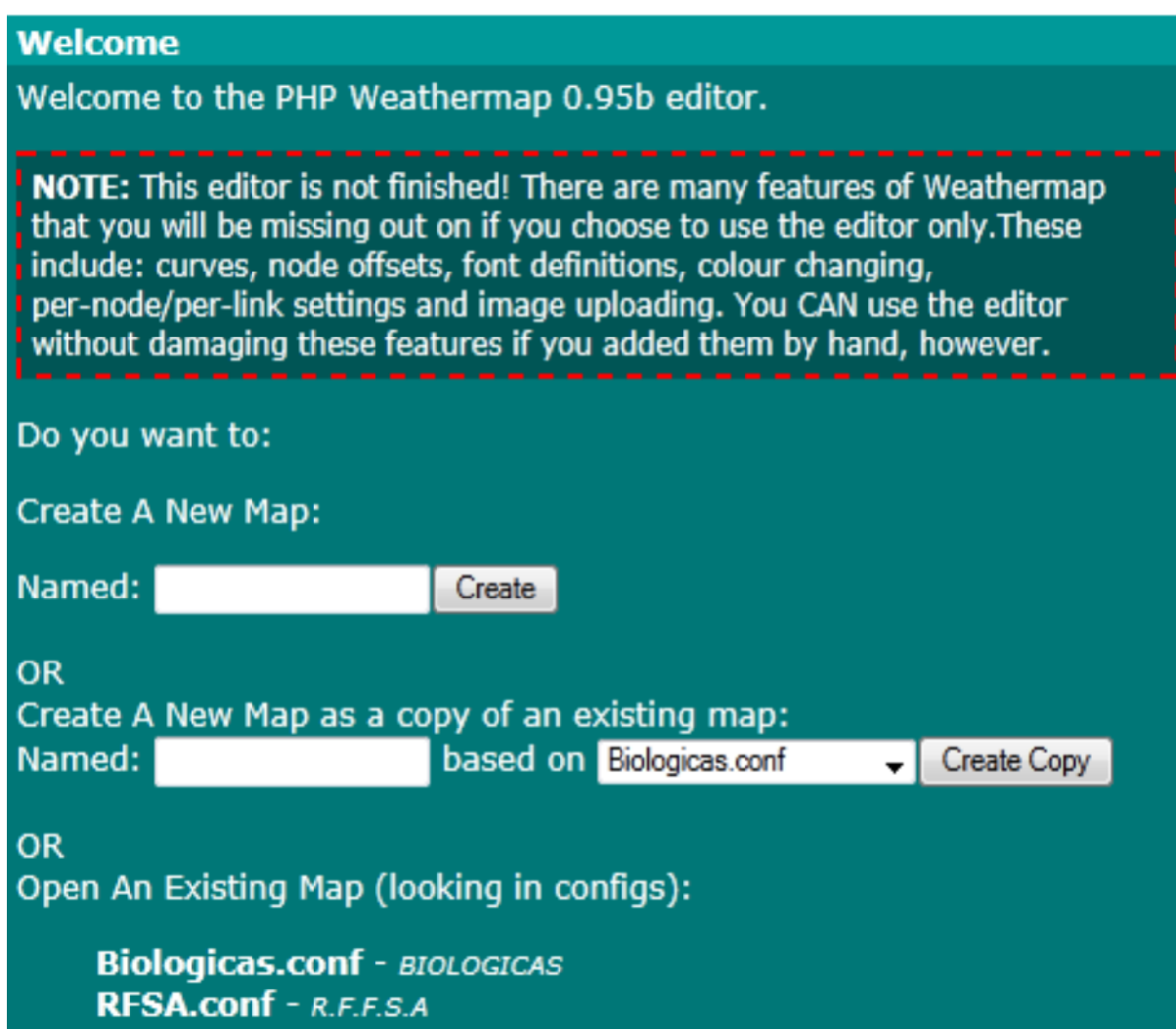


Figura 7 - Tela principal do Editor Gráfico (CCE, 2011)

O editor gráfico oferece três opções. Na primeira permite a criação de mapas totalmente novos, portanto, sem configuração alguma. Na segunda opção oferece a possibilidade de criar novos mapas, mas herdando as configurações de outro mapa. Esta é uma ótima opção para adicionar novos mapas e otimizar o trabalho, reutilizando configurações prévias do mapa modelo. E na terceira opção é possível editar os mapas já existentes.

Tanto para criar ou para modificar mapas, o editor gráfico permite as seguintes operações:

- Adicionar novos nós;
- Adicionar *links*;
- Alterar posicionamento da legenda do mapa (a legenda exibe o percentual de utilização dos *links*, através de escala de cores);
- Alterar a posição do *timestamp* (data e hora da última atualização);
- Alterar as propriedades do mapa. Elas englobam parâmetros como:
 - Título do mapa;
 - Texto da legenda;
 - Texto do *timestamp*;
 - Tamanho em *pixels* com que será desenhado o *link*;
 - Largura de banda do *link* (em bps – *bits* por segundo);
 - Dimensões da imagem de fundo do mapa;
 - Campo para fazer *link* com a imagem escolhida para fundo do mapa.
- Estilo do mapa. Tem como opções:
 - Rótulo dos *links* (Permite alterar a unidade de exibição da banda do *link*, entre: bps, percentual de utilização ou nenhum);
 - Estilo do HTML, *overlib* (dinâmico) ou estático;
 - Estilo para as setas dos *links*;
 - Opções de tipos e tamanhos de fontes para os nós, para os rótulos dos *links* e para a legenda.

Além das opções acima, é possível editar as propriedades individualmente de cada nó ou *link*, para isso basta clicar sobre um deles e serão exibidas telas como seguem nas Figuras 8 e 9, com as opções de propriedades para nós e *links* respectivamente.

Figura 8 – Exemplo de tela de propriedade para editar Nó (CCE, 2011)

Propriedade do Nó → Nas propriedades do Nó podemos alterar itens como Rótulo/Nome do Nó, imagem para ícone do Nó, mas a sua grande vantagem é poder inserir um *link* para outro Nó ou, até mesmo, outro mapa na opção “INFO URL”. Desta maneira, é criado o acesso a outros Nós ou mapas facilmente.

Figura 9 – Exemplo de tela de propriedade para editar Link (CCE, 2011)

Propriedade do *Link* → Aqui é possível alterar a banda do *link* e sua largura de exibição em *pixels*, mas o mais importante é através da opção “*PICK FROM CACTI*”, podemos direcionar o *link* para um gráfico específico, pois assim o usuário do mapa pode ter disponível através de apenas um clique os gráficos desejados, sem sair do mapa.

Estas são as opções disponíveis através do Editor Gráfico, as demais alterações somente serão possíveis através de um Editor de Texto do *Linux*, editando diretamente os arquivos de configuração correspondente a cada mapa.

3.1.2 Recursos disponíveis por meio do editor de texto

Utilizaremos o editor de texto *vi*, mas poderia ser qualquer outro do *Linux*, como por exemplo, *vim*, *lilo*, *nano*. O importante é que permita a edição de texto adequadamente.

A edição do arquivo de configurações do mapa do *PHP Network Weathermap*, não é ação trivial, há que se observar os parâmetros estabelecidos para cada função a ser implementada. Por isso, é recomendável observar as orientações que constam em seu manual (HOWARD, 2010-B), no qual está disponível, de forma detalhada, o processo correto para a configuração do recurso desejado.

A seguir, é exibida na figura 10 uma pequena parte de um arquivo de configuração, contendo textos explicativos sobre algumas configurações básicas do *PHP Network Weathermap*. Estes textos têm a finalidade de exemplificar a lógica de configuração adotada. No anexo deste trabalho, há maiores detalhes quanto ao arquivo de configurações, para quem deseja se aprofundar neste assunto.

```
# Este exemplo de arquivo de configuração, demonstra a maioria
dos recursos básicos do PHP Weathermap, juntamente com algumas
mudanças de layout. Segue abaixo descrição (em linha comentada - #)
e função a ser alterada:
# Exibe a imagem de fundo do mapa (nome do arquivo):
BACKGROUND background.png
# Exibe o título dado ao mapa:
TITLE Network Overview
# Mostra o estilo HTML escolhido:
HTMLSTYLE overlib
# Posição da legenda de cores dentro do mapa:
KEYPOS 10 400
# Definir algumas novas fontes TrueType para serem utilizadas no
mapa. Abaixo constam três fontes:
FONTDEFINE 100 VeraIt 8
FONTDEFINE 101 Vera 12
FONTDEFINE 102 Vera 9
#Mostra as configurações específicas para os links. Por exemplo,
BANDWIDTH é a largura de banda limite para o link, BWLABEL mostra os
valores da banda em bits, BWFONT e a fonte escolhida para os links e
a largura que o link é desenhado no mapa é de 4 pixels:
LINK DEFAULT
    BANDWIDTH 100M
    BWLABEL bits
    BWFONT 100
    WIDTH 4
```

Figura 10 – Exemplos de texto de arquivo de configuração (CCE, 2011)

Estes são alguns exemplos para ilustrar as configurações possíveis por meio do arquivo de configuração. Devemos observar que até mesmo as configurações executadas pelo Editor Gráfico também podem ser alteradas pela edição do arquivo. O editor de texto, portanto, permite o controle total de todas as configurações do *PHP Network Weathermap*.

3.1.3 Principais configurações

O arquivo de configurações é muito flexível, permite diversas combinações, basta alterar ou adicionar as funções desejadas. Por exemplo, é possível configurar os *links* como curvos, paralelos, com setas de sentido de fluxo, níveis de saturação através de escala de cores, com ou sem rótulo de banda, rótulo em percentual ou em bps.

A princípio, para um novo usuário, o manual do *PHP Network Weathermap* pode parecer complicado, mas com uma leitura das referências básicas é possível entender o método de configuração utilizado e, conseqüentemente, evoluindo o aprendizado. Seguem abaixo as principais possibilidades de configuração em cada item:

- Alterações nos Nós;
 - Mudar a fonte do rótulo;
 - Mudar a posição do nó;
 - Mudar o rótulo;
 - Mudar a cor do rótulo.
- Alterações nas setas dos *Links*;
 - Mudar o estilo das setas;
 - Mudar a largura com que a seta é desenhada;
 - Mudar a cor do contorno das setas.
- Alterações na imagem de fundo do Mapa;
 - Mudar a própria imagem;
 - Mudar a cor de fundo, caso não adicione nenhuma imagem;
 - Mudar a altura e largura do fundo, mas somente quando utilizar o fundo sem imagem.
- Alterações nos rótulos dos *Links*;
 - Alternar entre exibir os valores em *bits* por segundo ou percentagem de utilização;
 - Mudar a posição em que o rótulo é exibido no *link*. Muito útil para organizar grandes mapas;
 - Mudar a fonte do rótulo;
 - Mudar a cor do rótulo.

- Alterações na legenda de utilização dos *Links*;
 - É possível alterar o título e a posição da legenda;
 - Também é possível alterar o estilo de exibição da legenda. Por exemplo, na posição vertical ou horizontal;
 - Mudar o tamanho e a cor da fonte;
 - Mudar a cor da legenda;
 - É possível alterar as cores de exibição para cada nível percentual. Este recurso é extremamente útil, pois através das cores é possível alertar quanto ao nível de saturação dos links monitorados.
- Alterações nos Nós;
 - Pode-se inserir uma imagem como ícone para o nó, por exemplo, a imagem que represente um equipamento ativo naquele nó, um *switch*, um roteador;
 - Também é possível alterar a posição do rótulo do nó em relação à imagem inserida.
- Alterações nos *Links*;
 - Uma interessante possibilidade é a de construir *links* curvos, esta função é muito utilizada para representar vários *links* entre dois ou mais Nós, permitindo assim uma melhor visualização;
 - Outra interessante funcionalidade é a possibilidade de configurar vários *links* de modo que fiquem paralelos uns aos outros. Também, no caso de representar a ligação entre Nós, é bastante útil.
- Alterações na Legenda de Criação do Mapa;
 - É possível alterar tanto o seu formato quanto sua posição de exibição no mapa, como também a fonte utilizada.

A seguir, na Figura 11, são ilustradas de uma forma direta todas as possibilidades de alterações citadas acima. São também indicados quais os parâmetros de configuração que devem ser utilizados para cada item, facilitando assim a sua correta configuração.

How Do I Change That In Weathermap...?

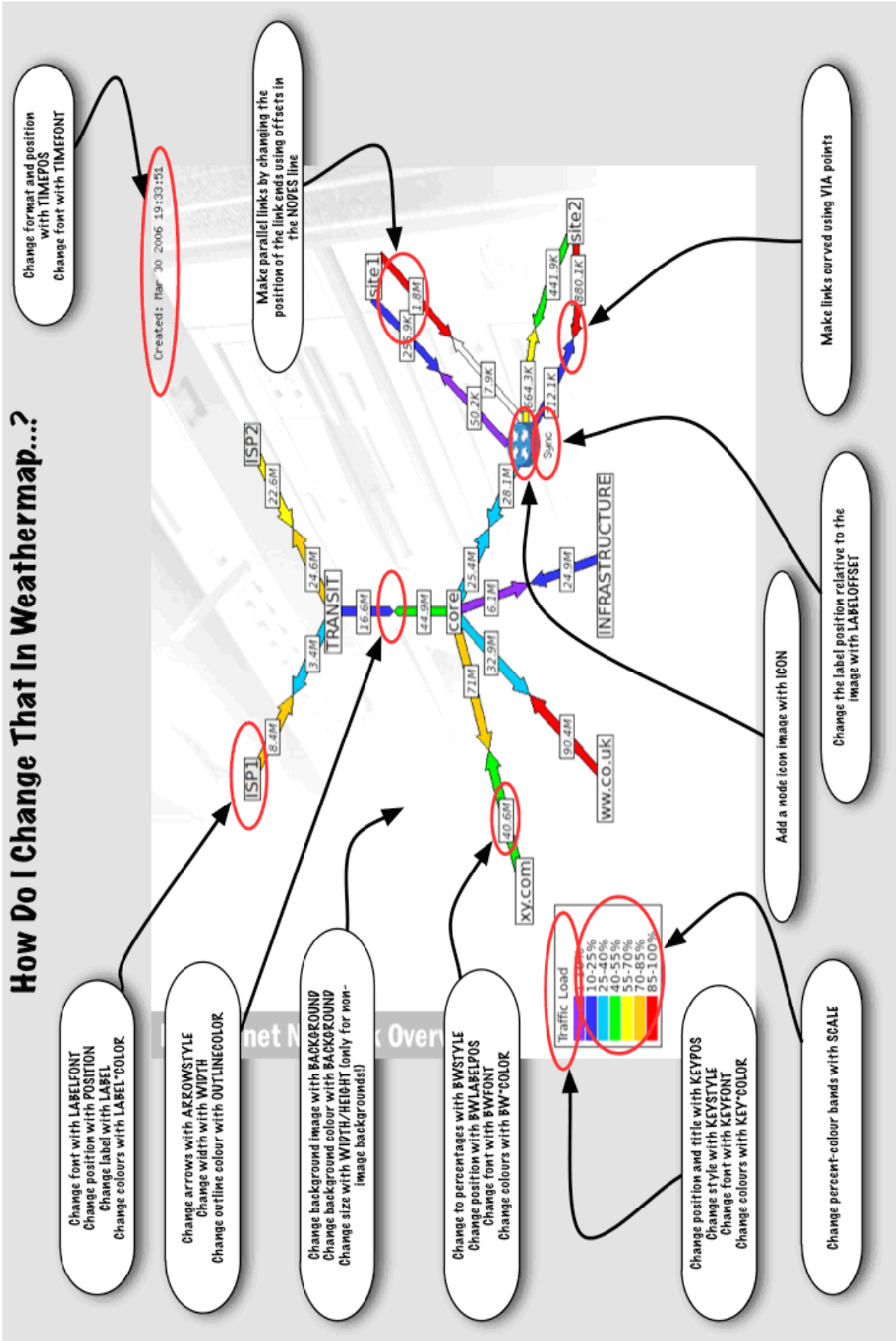


Figure 11 – Guia visual do PHP Network Weathermap (HOWARD, 2010)

3.1.4 Comparativo de demandas do ambiente monitorado

Para permitir uma visão mais precisa sobre que resultados foram obtidos com a utilização das ferramentas *Cacti* e *Weathermap* no monitoramento da rede da Universidade Federal do Paraná – UFPR, foi feita uma análise sobre as principais demandas que existiam antes da implantação do monitoramento e como elas foram atendidas após este processo. A seguir, a Tabela 1 apresenta o resultado quantitativo deste estudo.

Demandas	Antes	Depois
Alarme de falhas	Não era possível;	É possível recebe notificações por <i>E-mail</i> sobre equipamentos e hosts fora de operação;
Identificar <i>links</i> saturados	Não era possível;	O <i>PHP Network Weathermap</i> permite a rápida visualização de todos os <i>links</i> monitorados;
Visualização de toda rede	Não era possível;	Principal característica do <i>PHP Network Weathermap</i> permite a visualização de todos os Nós e <i>Links</i> da rede;
Localização rápida de falhas	Não era possível, somente através de reclamação de usuários;	A entrada em operação do monitoramento com <i>Cacti</i> e <i>PHP Network Weathermap</i> permitiu a ação pró-ativa, ou seja, agora é possível se antecipar ao usuário, localizando e resolvendo as falhas identificadas com muita rapidez;
Medir a largura de banda de um determinado <i>link</i>	Não era possível prontamente, somente através de acesso aos equipamentos envolvidos;	O <i>Cacti</i> apresenta gráficos que podem ser personalizados. Oferecendo opções com intervalos de minutos a meses;
Dimensionar <i>Links</i> e Redes	Não era possível;	Através dos mapas e gráficos é possível avaliar pontos de saturação na rede e redimensioná-los
Detectar anomalias no tráfego	Não era possível;	O <i>Cacti</i> juntamente com o <i>PHP Network Weathermap</i> oferece as primeiras informações, mas neste caso, são necessárias outras ações mais específicas, como por exemplo, a utilização de ferramentas que permitem a análise de tráfego, como o <i>NTOP</i> ou outro coletor de fluxo;

TABELA 1 – Comparativo antes e depois do monitoramento da rede UFPR

Para finalizar este capítulo, é oportuno apresentarmos outra opção de instalação e configuração do *Cacti* e seus principais *plugins*. Esta opção foi desenvolvida para facilitar e agilizar a utilização deste conjunto de ferramentas, é a instalação simplificada do *Cacti* denominada *CactiEZ*.

3.2 CACTIEZ

Uma das maiores limitações para que haja uma maior difusão, tanto do *PHP Network Weathermap*, como dos demais *plugins* do *Cacti*, tem sido a sua forma de distribuição, ou seja, apenas como *plugins*. Para o usuário final isto torna a instalação e configuração uma operação complexa e lenta. Para facilitar este processo, surgiu o *CACTIEZ* no início de 2009, disponibilizado pelo Desenvolvedor Jimmy Conner, um dos mais ativos colaboradores do projeto *Cacti*, com todos os seus principais *plugins* já instalados (CONNER, 2011-B).

Tudo é instalado por meio de um único pacote de dados. Esta opção foi criada visando justamente facilitar a instalação e configuração para o usuário final, o que permite, principalmente ao administrador de rede, ter em mãos uma ferramenta completa e sem dificuldades para sua implementação. Constam deste pacote *plugins* como: *Architecture*, *Syslog Collection*, *Reports*, *Auto Discovery*, *Router Config backup*, *Netflow Collection*, *PHP Network Weathermap*, e claro, o *Cacti*. Esta ferramenta está disponível no site: <http://cactiez.cactiusers.org/> em sua versão 0.6 (27/06/2011). É também um *software* livre e pode ser gravado em um único CD, o que permite uma instalação automatizada. Utiliza a distribuição *Linux CentOS 4.7*, mas segundo CONNER (2011-B), está em final de desenvolvimento a nova versão 0.7, que contará com várias outras facilidades, permitindo uma maior personalização, principalmente, para adicionar novos *plugins*, como também facilitando futuras atualizações. (CONNER, 2011-B).

No próximo capítulo trataremos da análise dos resultados deste trabalho.

4 ANÁLISE DOS RESULTADOS

Este capítulo apresenta os resultados das análises efetuadas na rede da Universidade Federal do Paraná, a partir da implantação do monitoramento com o Cacti e o *PHP Network Weathermap*.

4.1 DADOS OBTIDOS

Esta análise usou como base uma rede de grande porte, com mais de trezentos ativos gerenciáveis. Foram coletadas durante um período de trinta dias, todas as mensagens geradas pelo Cacti, conforme demonstrado na Tabela 2 a seguir:

ATIVOS	MENSAGENS	QUANTIDADE
329	Alarmes	176
	Retorno de Alarme	193
	Alertas	35
	Total	404

TABELA 2 – Dados analisados no Cacti

Para melhor entendimento, segue uma descrição de cada item da tabela 2:

- Ativos → São todos os equipamentos gerenciáveis disponíveis na rede em questão;
- Alarmes → Todas as mensagens recebidas via *e-mail* que informam a saída de operação de alguns dos dispositivos. Aqui vale ressaltar que em muitos casos, quando o equipamento apresenta falha intermitente, esta mensagem pode ser gerada várias vezes até que haja a solução do problema;
- Retorno de alarme → Mensagem que notifica o retorno da operação do Ativo que havia saído de operação;
- Alertas → São mensagens que avisam sobre limites mínimos ou máximos que podem ocasionar problemas na rede, como por exemplo, temperatura alta de um equipamento, *link* com percentual de saturação muito elevado por muito tempo, etc.

Estes dados permitem que se dimensione, o quão importante é para o administrador de rede contar com informações como estas, pois indicam exatamente os pontos da rede a serem rapidamente verificados.

A utilização destes dados em conjunto com a visualização dos mapas do *PHP Network Weathermap*, permite um controle, se não total, mas dentro de percentuais aceitáveis para o monitoramento de redes.

A centralização das informações em um único ponto possibilita a análise e a intervenção mais rápida, tudo isso se traduzindo em menor tempo de indisponibilidade da rede, maior economia de tempo e dinheiro, além da satisfação dos usuários.

4.2 PONTOS POSITIVOS

Entre os pontos positivos encontrados na utilização do *PHP Network Weathermap*, destacam-se:

- Permitir um monitoramento pró-ativo, pois de posse da informação no momento da ocorrência da falha, o administrador pode agir muito antes da solicitação do usuário, muitas vezes até sem que este perceba os efeitos do problema;
- A escala de cores disponível nos *links* exibidos pelo *PHP Network Weathermap* facilita a visão geral da rede, principalmente, destacando quando da ocorrência de saturação de *links*;
- Possibilita a visualização rápida de *links* ou equipamentos fora de operação;
- O acompanhamento da largura de banda dos *links*;
- Através da variação da largura de banda, auxilia a detectar tráfego malicioso;
- Visão global de toda rede, permitindo análise rápida;
- Oferece, através de seus históricos, a possibilidade de dimensionar pontos críticos, como por exemplo, *links* ou redes com o índice de saturação muito elevado;
- Possibilidade de configuração personalizada, flexibilizando e adequando seu uso as diversas topologias e situações.

4.3 PONTOS NEGATIVOS

Apesar de todas as vantagens identificadas, também observamos alguns pontos que podem ser melhorados ou até implementados:

- Processo de atualização para novas versões de *software*, pouco prático e que despende bastante tempo do usuário;
- Não disponibiliza todas as alterações de configuração possíveis através do Editor Gráfico, fato que torna lento, tanto a curva de aprendizado quanto o início efetivo de sua utilização;
- Ainda não disponível o envio de alarmes e alertas através de SMS ou outra plataforma móvel.

4.4 CONSIDERAÇÕES FINAIS

Por meio desta análise, concluímos que os pontos positivos encontrados são significativamente maiores que os pontos negativos, mesmo estes possuindo sua relevância, não impedem que esta ferramenta traga importantes benefícios ao gerenciamento de redes.

5 CONCLUSÃO

Este trabalho apresenta a ferramenta Cacti e seu *plugin PHP Network Weathermap*, traçando seu histórico, desenvolvimento e funcionalidades. Dá um enfoque do ponto de vista do administrador de rede, trazendo à tona características e recursos essenciais ao monitoramento de redes.

No cenário atual, nenhuma empresa ou corporação pode prescindir de um controle efetivo de sua rede, portanto, este trabalho traz a contribuição de selecionar e focar informações úteis para este segmento, de uma forma simples e prática, abrindo assim não somente a perspectiva de implementação das ferramentas citadas, mas de qualquer ferramenta de monitoramento. Consideramos que é importante despertar a necessidade de monitorar, buscar e propiciar constantemente o acesso a informação, com qualidade e rapidez. As principais características foram ilustradas com um caso prático de aplicação, tendo sido abordados pontos importantes da configuração bem como da apresentação do monitoramento de uma grande rede (UFPR-CCE).

Não obstante ainda constem pontos a serem aperfeiçoados, o *PHP Network Weathermap* demonstrou ser uma ferramenta de extrema utilidade para os profissionais da área de TI, onde seus pontos positivos superam com muita folga os pontos negativos. Com este trabalho, podemos observar as valiosas informações e o auxílio que ele pode prestar no gerenciamento dos diversos tipos de redes de computadores.

REFERÊNCIAS BIBLIOGRÁFICAS

CACTI, The Complete RRDtool-based Graphing Solution. **Cacti Manual 0.8.7**. Atualizada em Janeiro de 2011. Disponível em: <http://docs.cacti.net/manual:087>. Acesso em: 31 de Maio de 2011.

C C E, Centro de Computação Eletrônica da Universidade Federal do Paraná. Atualizada em Junho de 2011. Disponível em: <http://www.cce.ufpr.br>. Acesso restrito a utilização interna. Acesso em: 23 de Junho de 2011.

CHRISTIAS, Panagiotis. **Network Weathermap**. Atualizada em Fevereiro de 2006. Disponível em: <http://netmon.grnet.gr/weathermap/>. Acesso em: 07 de Junho de 2011.

CONNER, Jimmy. **CactiUser** – Plugins. Atualizada em Janeiro de 2011-A. Disponível em: <http://cactiusers.org/downloads/>. Acesso em: 07 de Junho de 2011.

CONNER, Jimmy. **CactiEZ**, Simplifying Cacti Installs. Atualizada em Maio de 2011-B. Disponível em: <http://cactiez.cactiusers.org/>. Acesso em: 07 de Junho de 2011.

COSTA, Felipe. **Ambiente de Rede Monitorado com Nagios e Cacti**. 1ª edição. Rio de Janeiro: Ciência Moderna, 2008.

HOWARD, Jones. **PHP Network Weathermap - See Everything, Documentation**. Atualizada em Março de 2010-A. Disponível em: <http://www.network-weathermap.com>. Acesso em: 14 de Junho de 2011.

HOWARD, Jones. **PHP Network Weathermap - See Everything, Documentation**. Atualizada em Março de 2010-B. Disponível em: <http://www.network-weathermap.com/manual/0.95b/>. Acesso em: 21 de Junho de 2011.

KUROSE, James F. e ROSS, Keith W. **Redes de Computadores e a Internet** – Uma abordagem Top Down. 3ª edição. São Paulo: Pearson Addison Wesley, 2006.

MATOS, Leonardo K. **Gerenciamento de equipamentos de rede utilizando o software CACTI**. Trabalho de Conclusão de Curso Especialização em Redes e Segurança de Sistemas. Pontifícia Universidade Católica do Paraná, 2009.

OETIKER, Tobias. **RRDtool** – Logging & Graphing, Documentation. Atualizada em Junho de 2011. Disponível em: <http://oss.oetiker.ch/rrdtool/doc/rrdtool.en.html>. Acesso em: 07 de Junho de 2011.

STALLINGS, William. **SNMP, SNMPv2, SNMPv3, RMON 1 and RMON 2**. 3ª edição. Massachusetts: Pearson Addison Wesley, 1999.

STALLINGS, William. **Redes e Sistemas de Comunicação de Dados**. 5ª edição. Rio de Janeiro: Campus, 2005.

SOURCEFORGE, The Project Cacti. Atualizada em Julho de 2010. Disponível em: <http://sourceforge.net/projects/cacti/>. Acesso em: 17 de Maio de 2011.

TANENBAUM, Andrew S. **Redes de Computadores**. 4ª edição. Rio de Janeiro: Campus, 2003.

ANEXO

CONFIGURAÇÃO MAPA PHP NETWORK WEATHERMAP

Segue na figura 12, exemplo de arquivo de configuração do *PHP Network Weathermap*, com as suas principais variações.

```
# Alguns comentários iniciais ...
#
# Este arquivo de configuração exemplo demonstra a maioria dos recursos básicos de
# PHP Weathermap, juntamente com algumas das mudanças cosméticas e layout possível
#
#
Background.png FUNDO
HTMLOUTPUTFILE example.html
IMAGEOUTPUTFILE example.png
TÍTULO Visão Geral da Rede
HTMLSTYLE overLib
KEYPOS 10 400

# Define algumas fontes TrueType novo - built-in os vão de 1 a 5, assim que
começar de alta
FONTDEFINE VeraIt 100 8
FONTDEFINE 101 Vera 12
FONTDEFINE 102 Vera 9

KEYFONT 102

PADRÃO-LINK
    BANDA 100M
    BWLABEL pedaços
    BWFONT 100
    OVERLIBWIDTH 395
    OVERLIBHEIGHT 153
    WIDTH 4

Nó Default
    LABELFONT 101

NODE trânsito
    POSIÇÃO 400 180
    TRÂNSITO LABEL

# Espirrar um pouco de cor de fundo para esses nós
NODE ISP1
    POSIÇÃO 250 100
    LABEL ISP1
    Infourl http://www.isp1.com/support/lookingglass.html
    LABELBGCOLOR 255 224 224

NODE isp2
    POSIÇÃO 550 100
    LABEL ISP2
    Infourl http://www.isp2.net/portal/
    LABELBGCOLOR 224 255 224
```

```

Núcleo NODE
    POSIÇÃO 400 300
    Núcleo LABEL
    Infourl https://core.mynet.net/admin/

NODE customer1
    Xy.com LABEL
    POSIÇÃO 150 370

NODE Customer2
    Ww.co.uk LABEL
    POSIÇÃO 250 450

NODE infra
    INFRA-ESTRUTURA LABEL
    POSIÇÃO 450 450

# Este nó tem um ícone, e assim nós empurramos o rótulo para a borda sul do mesmo,
por isso
# Ainda pode ser lido
NODE sync
    Sincronização LABEL
    ICON my_router.png
    LABELOFFSET S
    LABELFONT 2
    POSIÇÃO 550 370
# O ícone é tirado de um ícone Nagios pack:
#

NODE sitel
    LABEL sitel
    POSIÇÃO 700 220

NODE site2
    LABEL site2
    POSIÇÃO 750 420

Link sync-core
    NODES núcleo sync
    TARGET data/sync_traffic_in_259.rrd
    OVERLIBGRAPH
    Infourl
http://support.mynet.net/cacti/graph.php?rra\_id=all&local\_graph\_id=256
#
# Site1 tem dois Els, por isso usamos NODE-offsets para permitir que eles corram
paralelos
#

Link-sync sitela
    NODES sync: N sitel: W
    WIDTH 3
    TARGET data/sync_traffic_in_257.rrd
    BANDA 2M
    OVERLIBGRAPH
    Infourl
http://support.mynet.net/cacti/graph.php?rra\_id=all&local\_graph\_id=126
Link-sync sitelb
    NODES sync: E sitel: SE
    WIDTH 3
    OVERLIBGRAPH
    TARGET data/sync_traffic_in_258.rrd
    BANDA 2M
    Infourl
http://support.mynet.net/cacti/graph.php?rra\_id=all&local\_graph\_id=56

#
# Site2 também tem dois links, mas desta vez usamos o VIA para a curva de ligações
#

```

```

Link-sync site2a
  NODES sync site2
  WIDTH 3
  VIA 650 380
  TARGET data/sync_traffic_in_251.rrd
  OVERLIBGRAPH
  1M BANDWIDTH
  Infourl
http://support.mynet.net/cacti/graph.php?rra\_id=all&local\_graph\_id=252

Link-sync site2b
  NODES sync site2
  WIDTH 3
  VIA 650 420
  TARGET data/sync_traffic_in_252.rrd
  OVERLIBGRAPH
  1M BANDWIDTH
  Infourl
http://support.mynet.net/cacti/graph.php?rra\_id=all&local\_graph\_id=561

#
# 1 tem um ISP vários links, de novo, mas eles preferem ver uma flecha, e a
largura de banda agregada
# Por isso usamos múltiplos alvos em uma linha, aqui, para somar os dados

Link trânsito ISP1
  NODES trânsito ISP1
  TARGET data/trans1_traffic_in_352.rrd data/trans1_traffic_in_378.rrd
data/trans1_traffic_in_420.rrd
  BANDA 10M
  OVERLIBGRAPH
  Infourl
http://support.mynet.net/cacti/graph.php?rra\_id=all&local\_graph\_id=633

Link trânsito isp2
  NODES trânsito isp2
  TARGET data/trans1_traffic_in_438.rrd
  34m BANDWIDTH
  OVERLIBGRAPH
  Infourl
http://support.mynet.net/cacti/graph.php?rra\_id=all&local\_graph\_id=265

Core Link trânsito
  NODES núcleo de trânsito
  TARGET data/trans1_traffic_in_350.rrd
  ARROWSTYLE compacto
  WIDTH 4
  OVERLIBGRAPH
  Infourl
http://support.mynet.net/cacti/graph.php?rra\_id=all&local\_graph\_id=122

Link cust1-core
  NODES customer1 núcleo
  TARGET data/extreme_traffic_in_299.rrd
  OVERLIBGRAPH
  Infourl
http://support.mynet.net/cacti/graph.php?rra\_id=all&local\_graph\_id=237

Link cust2-core
  NODES núcleo Customer2
  TARGET data/extreme_traffic_in_286.rrd
  OVERLIBGRAPH
  Infourl
http://support.mynet.net/cacti/graph.php?rra\_id=all&local\_graph\_id=222

```

```

Link infra-core
NODES núcleo infra
TARGET data/extreme_traffic_in_294.rrd
OVERLIBGRAPH
Infourl
http://support.mynet.net/cacti/graph.php?rra_id=all&local_graph_id=228

```

Figura 12 – Exemplo de arquivo de configuração de mapa (CCE, 2011)

Na figura 13 é exibido o mapa resultante desta configuração:

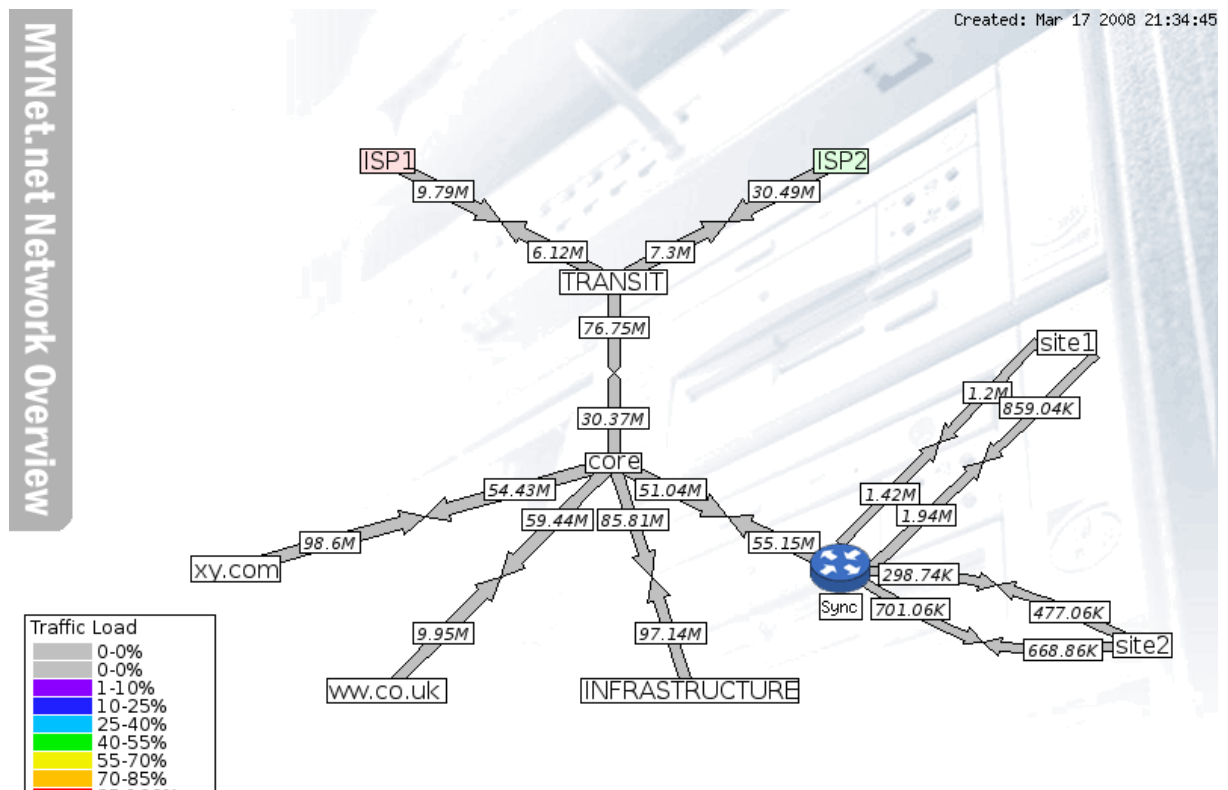


Figura 13 – Exemplo desenvolvido pelo autor do manual, Howard Jones. (HOWARD, 2010-B)